Econ 4567 Auction Theory: Caltrans Paper

Genevieve Mendoza and Claire Gottreich

May 3, 2023

This paper analyzes data from 705 procurement auctions held by the California Department of Transportation (commonly known as CalTrans).

1 Data

FIX!

1.1 Institutional Details

The California Department of Transportation (Caltrans) is the department that manages Aeronautics, Highway Transportation, Mass Transportation, Transportation Planning, Administration, and the Equipment Service Center in California. To outsource the labor for their highway construction projects, Caltrans runs low-bid procurement auctions. Within the auctions, there are typically large and small business bidders. Because small businesses have lower economies of scale, their costs are higher compared to large businesses.

Due to the difference in costs, Caltrans grants “bid preferences” to the small business bidders. Small businesses must meet three qualifications to be obtain the Small Business Certification. The certification requires that the business must be independently owned and operated in California, have no more than 100 employees, and over the last three tax years can only earn under $10 million average annual gross receipts. The benefit of the Small Business Certification is that there is a higher probability of winning the auction.

Once all bids are submitted, the lowest bid wins the auction. However, if a small business’s bid is within 5% of the lowest bid, they win the auction and are awarded the contract for construction. While the 5% discount is used to determine the winner, it is not applied to the actual amount the business is paid for the project: Caltrans will pay the true price the winner bid.

1.2 Data Overview

One immediate takeaway from the summary statistics is that small businesses are often refusing to bid on larger contracts, given that their bids are both much smaller on average and have a smaller standard deviation.

In addition, out of 705 auctions in total, there were 5.7 bidders in each auction on average, and the average bid was $968,241.65.

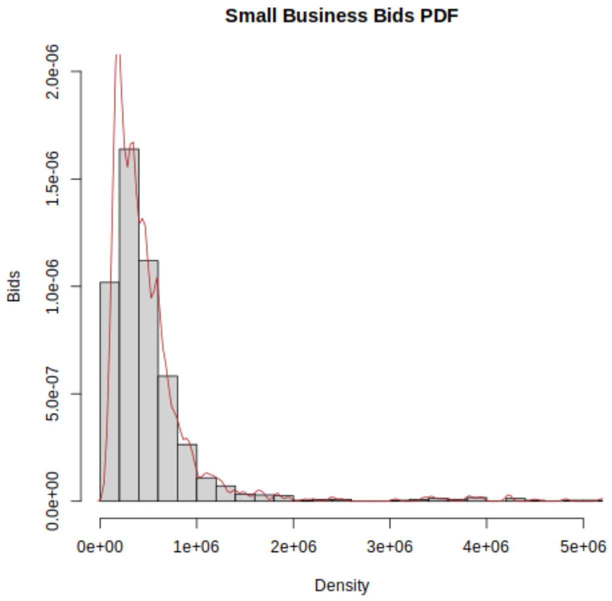
	Mean	Standard Deviation	Minimum	Maximum
Bids	986241.65	3311628.21	44655.00	58547700.00
Small Business Bids	531334.61	723168.10	49650.00	15485561.50
Number of Bidders	5.70	3.18	1.00	20.00
Business types present	1.64	0.48	1.00	2.00
Engineer's Estimates	943980.19	3734611.68	74000.00	60058000.00
Workdays	94.28	155.89	8.00	1430.00

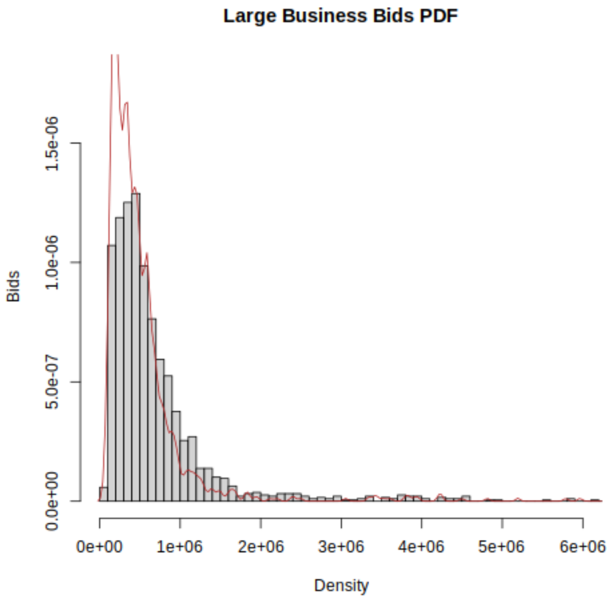
We calculated the winning bids and saw that on average, the winning bid was $39,417 lower than the state's cost estimate. This is a relatively small difference, but it does suggest that competition among the bidders does drive the procurement cost down for the state. It could also potentially represent a winner's curse if the cost estimates are very accurate, but this would require more analysis to confirm. Furthermore, the standard deviation of the differences is quite high at almost $1 million, and there are some cases where Caltrans is forced to pay a premium over their estimate - it is unclear if this represents unreliability by the engineers making the estimates, or an underlying dynamic of the auction process.

1.3 Bidding Behavior

1.3.1 Kernel Density Estimation

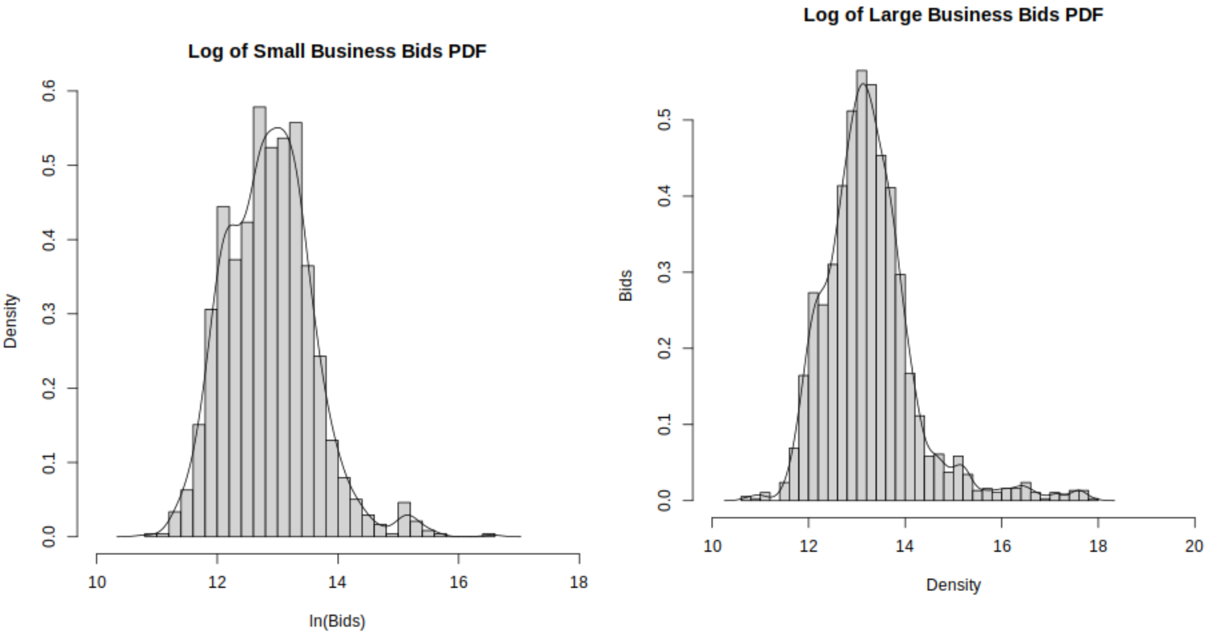
First we estimated the true distribution of bids for large and small businesses using kernel density estimation with a bandwidth selected via leave-one-out cross-validation. Kernel density estimation relies on the second derivative of the density function, and will as a result tend to overestimate peaks and underestimate valleys. We can see that in this plot, where the density estimate is much higher than the first peak.





The large business bids do have a longer tail, which reflects that they are more able to take on projects with a large cost than the smaller businesses. There are also clusters in the bid frequencies, which we can see in the spikes in the KDEs of the PDFs. It makes sense that the distribution of bids is not perfectly uniform, but instead clusters because projects can be grouped into similar size categories, and because people generally prefer to deal with round numbers when it comes to money. Many more projects will receive bids at $1 million than at $1,010,000.

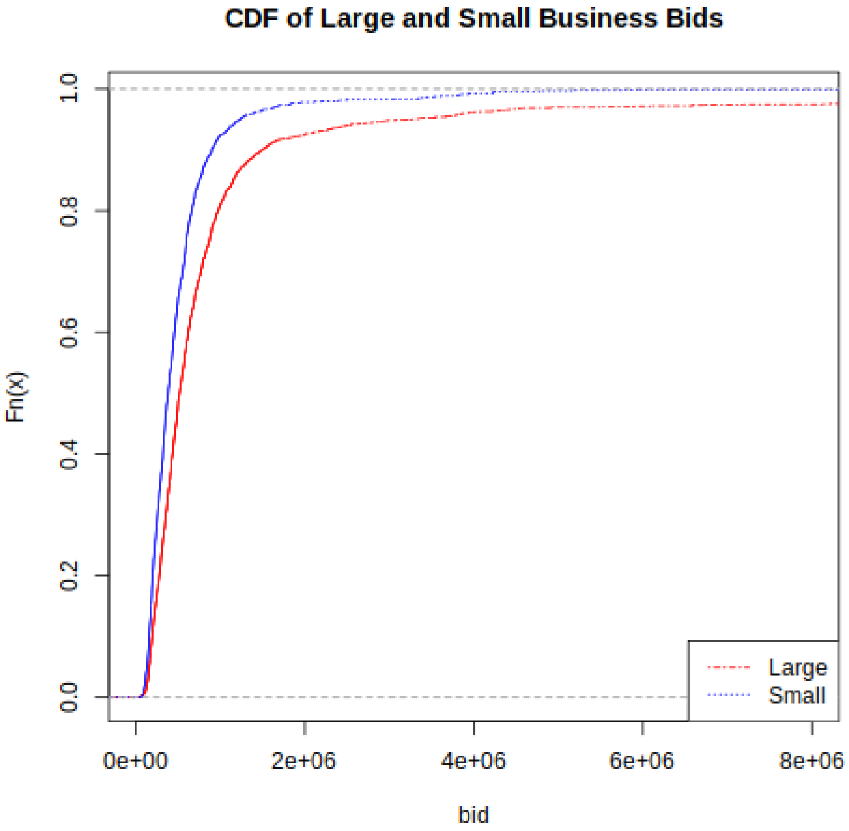
Because of the data's skew, we applied a log transformation to normalize the data. Log scaling data can effectively compress very wide ranges down, so that extreme bids are proportionally less large. Consequently, the goal of this transformation was to provide a better visualization of the long tail in our data. This reveals a very slight bimodal peak around bids of about $3 million (exp(15)), which might be a common project cost. The larger businesses look to have a very similar distribution, but just have a higher mean, and their tail also extends further up to bids of around $7-8 million.



(a) PDF of the log-bids by small businesses

(b) PDF of the log-bids excluding small businesses

We also plotted the CDFs to see patterns in the bid sizes between the two types of bidders. The smaller average bids of the small businesses are very clear in this plot, as the line is strictly greater than the large businesses' estimated CDF.



1.3.2 Regressions

To better understand the bidding behavior of business, we ran regressions on bidders, engineer's estimate, and work days. Bidders is how many bidders there were for a certain project, engineer's estimate is the estimate on how much a professional believes the project costs, and work days is how long the project will take. The first regression we ran included both small and large businesses, and below is the result.

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	222576.4332	32736.1113	6.80	0.0000
num_bidders	-18609.7317	4701.4001	-3.96	0.0001
Estimate	0.8493	0.0042	203.89	0.0000
WorkDays	718.0144	100.7270	7.13	0.0000

The above table shows the output of three regressions of the number of bidders, estimated cost, and workdays spent on the job, on the procurement bids. The first covers all of the data, the second only certified small businesses, and the third large businesses. The results of this regression are as we expected. The negative sign on bidders is consistent with the theory we studied, because with more bidders in a particular auction, it drives down the price of the bid. Similarly, as the engineer's estimate increases, the bid should increase because there is a higher cost for the project; and as work days increase, so should the bid because projects that take longer become more expensive.

The results for small businesses also have the same signs as we expected, for the same reasons as above. The only difference in this regression is the degree to which work days

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	222576.4332	32736.1113	6.80	0.0000
num_bidders	-18609.7317	4701.4001	-3.96	0.0001
Estimate	0.8493	0.0042	203.89	0.0000
WorkDays	718.0144	100.7270	7.13	0.0000

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	222576.4332	32736.1113	6.80	0.0000
num_bidders	-18609.7317	4701.4001	-3.96	0.0001
Estimate	0.8493	0.0042	203.89	0.0000
WorkDays	718.0144	100.7270	7.13	0.0000

influences the bid. The coefficient for the regression with only small business bidders is 154.3 compared with 716 for the regression including all businesses. This might indicate that number of work days does not increase the bid as much because smaller businesses might not be able to handle the cost of larger projects that require more time and capital. Therefore, they can't bid as high.

A notable difference in the final regression, which is limited to businesses that tend to be larger, is the greater coefficient for work days, such that there is a much larger coefficient on work days for large business bidders. This can be explained by the fact that large businesses can handle the higher costs of lengthier projects. Their economies of scale allow for them to afford projects that require more time and capital. Because of this, they are more likely to pursue such projects and can bid higher for them.

2 Model

2.1 Assumptions

First, recall the auction rules: a small business bid within 5% of the lowest bid will win, and the procurer will receive their bid. Thus, the 5% rule affects the expectation of winning, but not the payoff. Because there are two groups of businesses, we have asymmetric bidders which we will refer to as L and S, with $n_L + n_S = n$, and $n \geq 2$. Similarly, refer to the CDFs of the costs as $F_L \neq F_S$ on the support $[\underline{c}, \bar{c}]$.

We assume that during the CalTrans auction, the bidders had costs that were independent and drawn from an identical distribution, and that the costs were private. (Since these were procurement auctions, cost is the analogue to valuation). However, the distribution is conditional on some estimate.

These distributions are unknown, and we do not have complete information, on their parameters. However, we do know that they are conditional on an engineer's estimate provided by CalTrans, which approximates the true cost. distributed according to some unknown, distinct parameters, Therefore, we write that $c \sim F_{L \text{ or } S}(\cdot \mid E)$, where $\cdot$ represents parameters and E represents the estimate.

Bidding functions are assumed to be monotonically increasing in cost. Similarly, all bidding functions are assumed to satisfy $\beta(0) = 0$. Require that $\beta_L(\bar{c}) = \beta_S(\bar{c})$, and that each β_i has an inverse $\phi_i = \beta_i^{-1}$.

Finally, we assume that within the groups, each bidder uses the same optimal strategy, denoted β_S and β_L.

2.2 Bidders' Optimization Problems

Since the auction is a first price auction, we can define the expected payoffs for a small business i (with large business competitors k and other small business competitors $j \neq i$, where the numbers of small/large bidders are n_S, n_L) as:

$$P(b_i \text{ wins}) = P\left(b_i < b_j \forall j \neq i \wedge \frac{1}{1.05} b_i < b_k \forall k\right)$$

$$\Pi(\beta_S(c_i), c_i) = \Pi(b_i, c_i) = (b_i - c_i) \cdot P\left(b_i < \beta_S(c_j) \forall j \neq i \wedge b_i < 1.05 \beta_L(c_k) \forall k\right)$$

Since valuations are private, denote opponent bids as b_j, b_k

$$\begin{aligned} \Pi(b_i, c_i) &= (b_i - c_i) \cdot P\left(b_i < b_j \forall j \neq i \wedge b_i < 1.05 b_k \forall k\right) \\ &= (b_i - c_i) \cdot \left(\left(1 - F_S(\beta_S^{-1}(b_i) \mid E)\right)^{n_S - 1} \left(1 - F_L(\beta_L^{-1}(1.05 b_i) \mid E)\right)^{n_L}\right) \end{aligned}$$

Similarly, for a large business i with bid $b_i = \beta_L(c_i)$, with small businesses j and other large busineses $k \neq i$:

$$\begin{aligned} P(b_i \text{ wins}) &= P\left(b_i < \frac{1}{1.05} \cdot b_j \forall j \wedge b_i < b_k \forall k \neq i\right) \\ \Pi(\beta_L(c_i), c_i) &= (b_i - c_i) \cdot P\left(b_i < \frac{1}{1.05} \cdot b_j \forall j \wedge b_i < b_k \forall k \neq i\right) \\ &= (b_i - c_i) \cdot \left(\left(1 - F_S(\beta_S^{-1}(\frac{b_i}{1.05}) \mid E)\right)^{n_S} \left(1 - F_L(\beta_L^{-1}(b_i) \mid E)\right)^{n_L - 1}\right) \end{aligned}$$

Then, they will choose their bid based on their first order condition, $\max_{b_i} \Pi(b_i, c_i)$ or $\max_{b_i} \Pi(b_i, c_i)$.
For a small business, this gives the differential equation:

$$1 = (b_i - c_i) \cdot \left(\frac{(n_S - 1) \cdot f_S(c_i \mid E)}{1 - F_S(c_i \mid E) \cdot \beta_S'(c_i)} + \frac{(n_L) \cdot f_L(1.05 c_i \mid E)}{1 - F_L(1.05 c_i \mid E) \cdot \beta_L'(c_i)}\right)$$

Similarly, for a large business the first order condition gives:

$$1 = (b_i - c_i) \cdot \left(\frac{(n_S) \cdot f_S(\frac{1}{1.05} c_i \mid E)}{1 - F_S(\frac{c_i}{1.05} \mid E) \cdot \beta_S'(c_i)} + \frac{(n_L - 1) \cdot f_L(c_i \mid E)}{1 - F_L(c_i \mid E) \cdot \beta_L'(c_i)}\right)$$

FIXME: the boundary conditions need to take the 5% rule into account.
Together we have a system of equations satisfying $\beta_S(\bar{c}) = \beta_L(\bar{c}) = \bar{c}$ and $\beta_S(\underline{c}) = \beta_L(\underline{c})$. From this system we will be able to obtain the costs and $F_{L \text{ or } S}(\cdot \mid E)$.

3 Identification

3.1 Empirical Strategy

While we have all the submitted bids, we do not have the original costs businesses estimated for themselves. However, assuming all bidders used an optimal strategy, we can

recover those costs.
Recall the FOCs and differential equations:

$$
\begin{aligned}
&\max_{b_i} \Pi(b_i, c_i), \qquad \max_{b_i} \Pi(b_i, c_i) \\
&1 = (b_i - c_i) \cdot \left(\frac{(n_S - 1) \cdot f_S(c_i \mid E)}{1 - F_S(c_i \mid E) \cdot \beta_S'(c_i)} + \frac{(n_L) \cdot f_L(1.05c_i \mid E)}{1 - F_L(1.05c_i \mid E) \cdot \beta_L'(c_i)} \right) \\
&1 = (b_i - c_i) \cdot \left(\frac{(n_S) \cdot f_S(\frac{1}{1.05}c_i \mid E)}{1 - F_S(\frac{c_i}{1.05} \mid E) \cdot \beta_S'(c_i)} + \frac{(n_L - 1) \cdot f_L(c_i \mid E)}{1 - F_L(c_i \mid E) \cdot \beta_L'(c_i)} \right)
\end{aligned}
$$

These have the boundary conditions $\beta_S(\bar{c}) = 1.05\beta_L(\bar{c}) = 1.05\bar{c}$ and $\beta_S(\underline{c}) = 1.05\beta_L(\underline{c})$. Just as f is the pdf of costs, denote the pdf of bids by g.

Now recall that $G(b; E, n_S, n_L$ is the CDF of bids (with parameters: engineer's estimate, number of small businesses, and number of large businesses) and $F(c; E, n_S, n_L)$ is the CDF of costs, with costs $c_i = \beta^{-1}(b_i)$ for the appropriate β_S, β_L. Furthermore, bidding functions are strictly monotonic and therefore one-to-one (map one cost uniquely to one bid). Given this, a lower bid implies a firm also has a lower cost. We show that for one bidder's bid B:

$$
\begin{aligned}
1 - G_S(b_i \mid n_S, n_L, E) &= P(\beta_S^{-1}(B) \geq \beta_S^{-1}(b) \mid E) \text{ because bids are monotonic} \\
&= P(C \geq B_S^{-1}(b) \mid E) = 1 - F_S(\beta_S^{-1}(b)) \\
G_S(b_i \mid n_S, n_L) &= F_S(\beta_S^{-1}(b_i))
\end{aligned}
$$

and

$$
\begin{aligned}
1 - G_L(b_i \mid n_S, n_L, E) &= P(\beta_L^{-1}(B) \geq \beta_L^{-1}(b) \mid E) \text{ because bids are monotonic} \\
&= P(C \geq B_L^{-1}(b)) = 1 - F_L(\beta_L^{-1}(b)) \\
G_L(b_i \mid n_S, n_L) &= F_L(\beta_L^{-1}(b_i)).
\end{aligned}
$$

Furthermore, to get g_S and g_L we must take the derivatives of G_S and G_L: $\frac{dx}{dy} = \frac{1}{m'(x)} \implies \frac{dm^{-1}}{dy} = \frac{1}{m'(m^{-1}(y))}$:

$$
\begin{aligned}
\frac{d}{db_i} G_L(b_i \mid n_S, n_L, E) &= \frac{d}{db_i} F_L(\beta_L^{-1}(b_i)) \\
g_L(b_i \mid n_S, n_L, E) &= f_L(\beta_L^{-1}(\beta_L(c_i)))\beta_L'(\beta_L^{-1}(b_i))
\end{aligned}
$$

Similarly,

$$
F_S'(\beta_S^{-1}(b)) = f_S(\beta_S^{-1}(b)) \cdot \frac{1}{\beta_S'(\beta_S^{-1}(b))}
$$

3.1.1 Recovering Bidder Costs

Plugging this in to the partial differential equation which gives the optimal bidding function for small businesses, we obtain:

$$
\begin{aligned}
1 &= (b_i - c_i)\cdot \left(\frac{(n_S - 1) \cdot g_S(\beta_S(c_i \mid E) \mid n_S, n_L) \cdot \beta_S'(\beta_1^{-1}(b)}{1 - G_S(b_i \mid n_1, n_2) \cdot \beta_S'(c_i)} + \frac{(n_L) \cdot g_L(1.05 b_i \mid n_L, n_S) \cdot \beta_L'(c_i)}{1 - G_L(1.05 b_i \mid n_L, n_S) \cdot \beta_L'(c_i)} \right) \\
&= (b_i - c_i)\cdot \left(\frac{(n_S - 1) \cdot g_S(\beta_S(c_i \mid E) \mid n_S, n_L)}{1 - G_S(b_i \mid n_1, n_2)\cdot} + \frac{(n_L) \cdot g_L(1.05 b_i \mid n_L, n_S)}{1 - G_L(1.05 b_i \mid n_L, n_S)} \right) \\
c_i(b_i; E, n_S, n_L) &= b_i - \left(\frac{(n_S - 1) \cdot g_S(\beta_S(c_i \mid E) \mid n_S, n_L)}{1 - G_S(b_i \mid n_1, n_2)\cdot} + \frac{(n_L) \cdot g_L(1.05 b_i \mid n_L, n_S)}{1 - G_L(1.05 b_i \mid n_L, n_S)} \right)^{-1}
\end{aligned}
$$

Similarly, we can put the costs in terms of bids and g_L for large businesses:

$$
\begin{aligned}
1 &= (b_i - c_i)\cdot \left(\frac{(n_S) \cdot g_S(\frac{1}{1.05}\beta_S(c_i \mid E) \mid n_S, n_L) \cdot \beta_S'(\beta_1^{-1}(b)}{1 - G_S(\frac{1}{1.05} b_i \mid n_1, n_2) \cdot \beta_S'(c_i)} + \frac{(n_L - 1) \cdot g_L(b_i \mid n_L, n_S) \cdot \beta_L'(c_i)}{1 - G_L(b_i \mid n_L, n_S) \cdot \beta_L'(c_i)} \right) \\
&= (b_i - c_i)\cdot \left(\frac{(n_S) \cdot g_S(\frac{1}{1.05}\beta_S(c_i \mid E) \mid n_S, n_L)}{1 - G_S(\frac{1}{1.05} b_i \mid n_1, n_2)\cdot} + \frac{(n_L - 1) \cdot g_L(b_i \mid n_L, n_S)}{1 - G_L(b_i \mid n_L, n_S)} \right) \\
c_i(b_i; E, n_S, n_L) &= b_i - \left(\frac{(n_S) \cdot g_S(\frac{1}{1.05} b_i \mid n_S, n_L)}{1 - G_S(\frac{1}{1.05} b_i \mid n_1, n_2)\cdot} + \frac{(n_L - 1) \cdot g_L(b_i \mid n_L, n_S)}{1 - G_L(b_i \mid n_L, n_S)} \right)^{-1}
\end{aligned}
$$

In the next section, we can then use the bid data and kernel density estimates of G_S, G_L to recover estimates of the cost. The goal of this section was to recover estimated costs for each bidder.

To simplify the conditional statements that we will have to make, we will only focus on the most common auctions that have a particular pair of number of small and large businesses. The largest sample in the Cal Trans data set is one small business and three large businesses. Focusing on only one pair of n_S and n_L simplifies the calculation of density functions.

As we found in the previous section, for small businesses:

$$
c_i(b_i; E, n_S, n_L) = b_i - \left(\frac{(n_S - 1) \cdot g_S(\beta_S(c_i \mid E) \mid n_S, n_L)}{1 - G_S(b_i \mid n_1, n_2)\cdot} + \frac{(n_L) \cdot g_L(1.05 b_i \mid n_L, n_S)}{1 - G_L(1.05 b_i \mid n_L, n_S)} \right)^{-1}
$$

and for large businesses:

$$
c_i(b_i; E, n_S, n_L) = b_i - \left(\frac{(n_S) \cdot g_S(\frac{1}{1.05} b_i \mid n_S, n_L)}{1 - G_S(\frac{1}{1.05} b_i \mid n_1, n_2)\cdot} + \frac{(n_L - 1) \cdot g_L(b_i \mid n_L, n_S)}{1 - G_L(b_i \mid n_L, n_S)} \right)^{-1}
$$

In these equations, g_s, G_s, etc., represent the pdf and cdf of the bids. To find them, we use the definition of conditional probability:

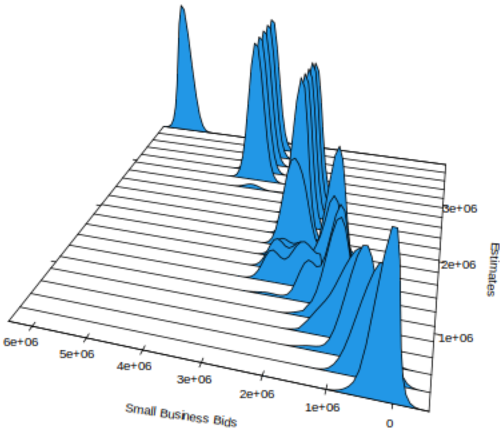
$$
P(A \mid B) = \frac{P(A \wedge B)}{P(B)}
$$

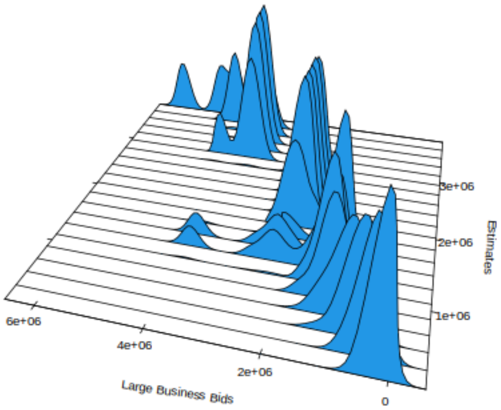
Therefore, we need to estimate densities for the joint probability of bids and estimates and the marginal probability of estimates.

We will first estimate functions for the bids (i.e., g_s and g_l in the above equation). They will be conditional on the engineers estimate. To simplify this, we will condition on the median of the engineers estimate. For our subset of the auctions with the most common combination of large/small bidders, this was \$526000.

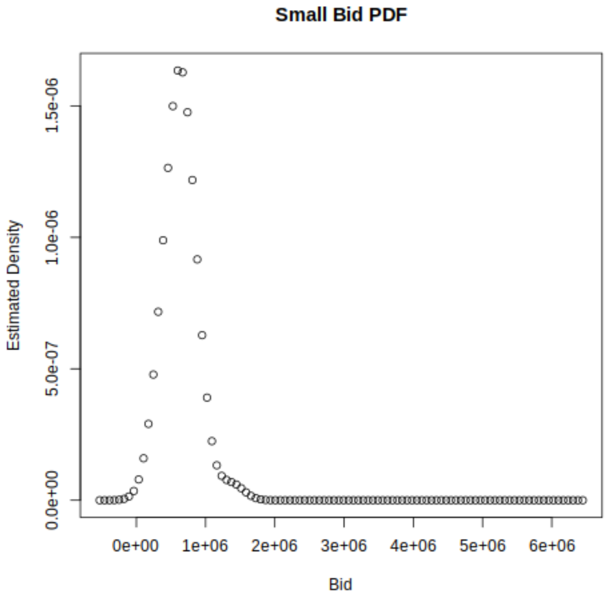
We will now estimate the PDF of small business bids which is conditional on the median of the engineers estimate. It will not be conditional on the number of small and large businesses because we are fixing it to our chosen subset of auctions.

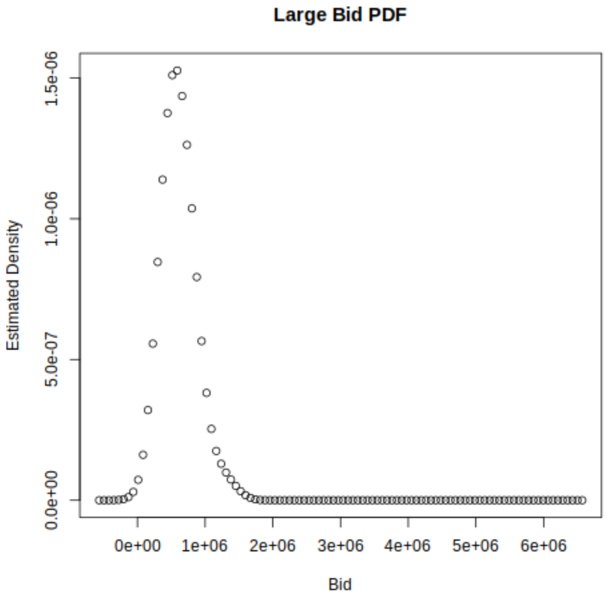
First, here are 3D plots of the conditional PDFs of bids on estimates, for small and large businesses.





Then, here are the graphs of the PDF for small/large businesses evaluated on their bids, conditional on the median estimate from our subset (which is the same for both large and small businesses, given they compete in the same auctions).

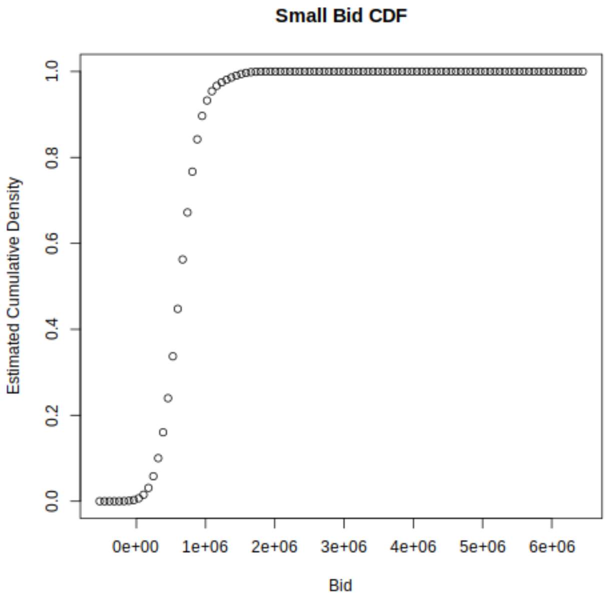


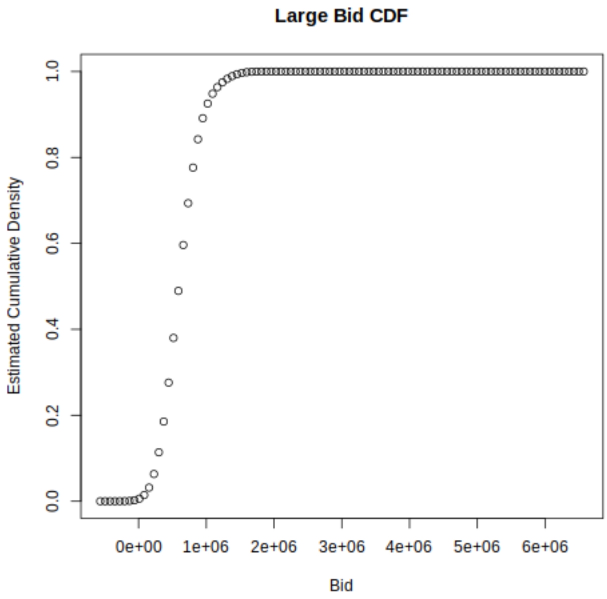


While it is counterintuitive that the small businesses actually have a mode slightly to the right of the large businesses, in our subset of the data the median small business bid was \$615605 and the median large business bid was \$585513. The large businesses do have a handful of bids on much larger projects (up to 50 million dollars in the full data set), but they are not present in our subset of the data.

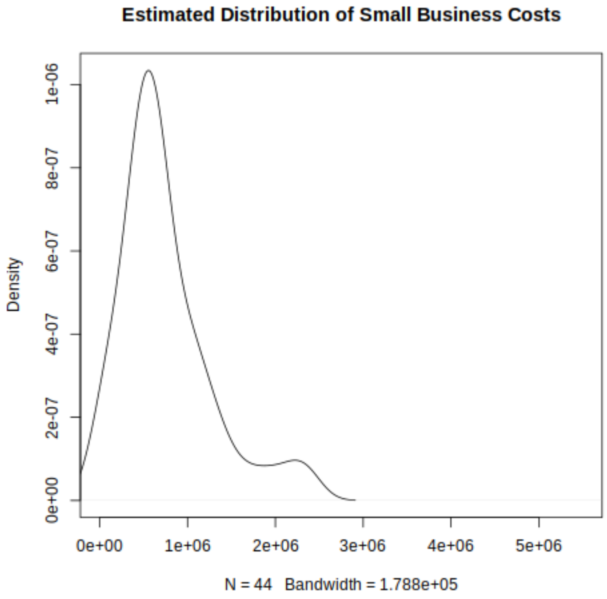
Even within our subset, the number of large business bids on larger projects is small enough to not be very visible on the density plots. However, once we recover the costs, it becomes more noticeable.

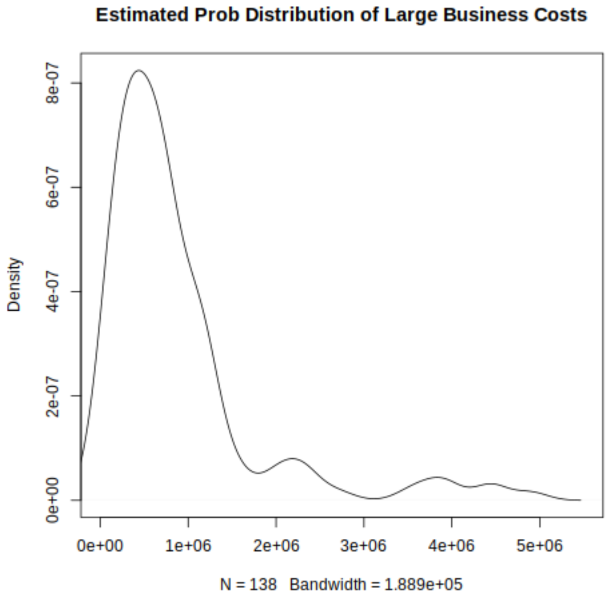
Similarly, below are the two CDFs of bids ($\boldsymbol{G_s}$ and $\boldsymbol{G_l}$ in the above equation).

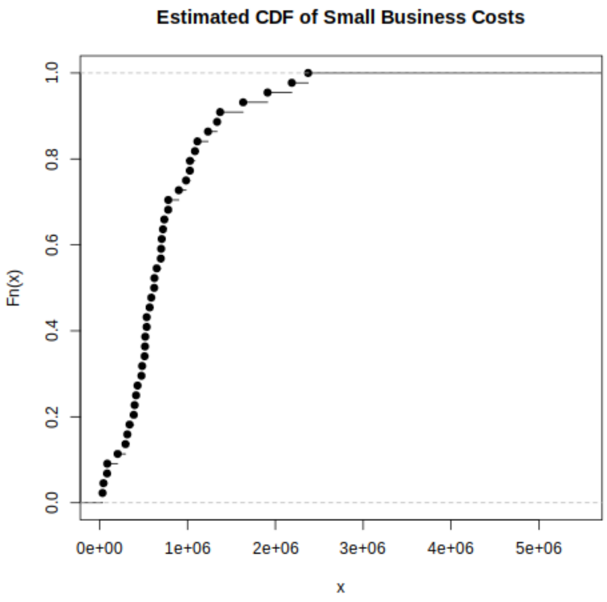


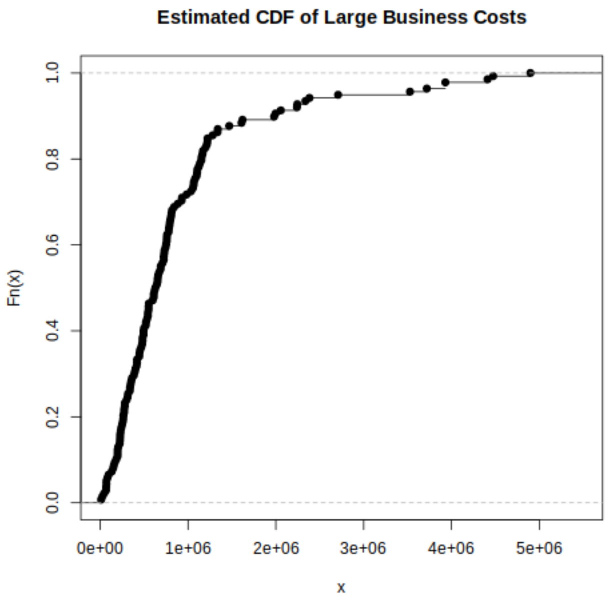


Finally, we calculate the estimated cost distribution by recovering the costs from the bid distribution, using the equation above (calculated in the Identification section).
The scale of these plots was set to $6 million for both small and large businesses: the reason the small business plots stop much earlier is because they generally did not participate in auctions for the very large projects, but rather in auctions where their costs would be smaller (typically under a million dollars).






Estimated CDF of Small Business Costs
Fn(x)
0.0
0.2
0.4
0.6
0.8
1.0
0e+00
1e+06
2e+06
3e+06
4e+06
5e+06
x


Estimated CDF of Large Business Costs
Fn(x)
0.0
0.2
0.4
0.6
0.8
1.0
0e+00
1e+06
2e+06
3e+06
4e+06
5e+06
x

The code used to generate these graphics is below.

```
# n by 2 matrix of n_S and n_L for each i in n
dt_bids <- cbind(calt_full$NumberofSmallBusinessBidders,
                 calt_full$NumberofLargeBusinessBidders)
rows <- c()
for (row_i in seq_len(nrow(dt_bids))) {
  current_row <- dt_bids[row_i, ]
  row_as_char <- paste(current_row, collapse = " ")
  rows <- c(rows, row_as_char)
}
sort(table(rows), decreasing = TRUE)
calt_subset <- calt_full[calt_full$NumberofSmallBusinessBidders ==
   1 &
                         calt_full$NumberofLargeBusinessBidders ==
                            3, ]

# vectors of large/small bids from our subset
sb_bids_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   1, ]$Bid
lb_bids_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   0, ]$Bid
# change these further down
ests_subset <- calt_subset$Estimate
# there is just one estimate per auction, so median estimate is
   same for large/small
median_estimate <- median(ests_subset)
med_est_vec <- rep(median_estimate, length(sb_bids_sub)) # and lb
   bids too

sb_ests_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   1, ]$Estimate
# this is just sb_ests_sub with each entry repeated three times
lb_ests_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   0, ]$Estimate

# repeat the median appropriate number of times
sb_med_est_vec <- rep(median_estimate, length(sb_bids_sub))
lb_med_est_vec <- rep(median_estimate, length(lb_bids_sub))

library(hdrcde)
library(devtools)
devtools::install_github("https://github.com/sethmcg/climod")
library(climod)

# cde(x,y) gives p(y|x)
ests_grid <- seq(from = min(sb_ests_sub), to = max(sb_ests_sub),
   length = 21)
g_l <- cde(lb_ests_sub, lb_bids_sub, deg = 1, link = "log",
   nxmargin = 21,
           x.name = "Estimates", y.name = "Large Business Bids")
```

```
png("./src/imgs/g_l_cond.png")
plot(g_l)
dev.off()

# y is a grid of length 100 plugged in for bids
# z has 100 columns: the third row is those evaluated at the median
   estimate
png("./src/imgs/g_l_median.png")
plot(g_l$y, g_l$z[3, ]) # at the median estimate
dev.off()
g_s <- cde(sb_ests_sub, sb_bids_sub, deg = 1, link = "log",
   nxmargin = 21,
           x.name = "Estimates", y.name = "Small Business Bids")
png("./src/imgs/g_s_cond.png")
plot(g_s)
dev.off()
png("./src/imgs/g_s_median.png")
plot(g_s$y, g_s$z[3, ]) # at the median estimate
dev.off()

# we're just taking part of joint PDF, but pdf2cdf can normalize to
   1 so it's ok
G_l <- pdf2cdf(g_l$z[3, ], g_l$y)
png("./src/imgs/G_l.png")
plot(G_l)
dev.off()
G_s <- pdf2cdf(g_s$z[3, ], g_s$y)
png("./src/imgs/G_s.png")
plot(G_s)
dev.off()

g_s_spline <- splinefun(g_s$y, g_s$z[3, ])
# integrate(g_s_spline, 100, 6e6) gives 0.996
g_l_spline <- splinefun(g_l$y, g_l$z[3, ])
g_s_s <- g_s_spline(sb_bids_sub)
g_l_l <- g_l_spline(lb_bids_sub)
g_l_s_105 <- g_l_spline(1.05 * sb_bids_sub)
g_s_l_105 <- g_s_spline(lb_bids_sub / 1.05)

G_s_spline <- splinefun(G_s)
G_l_spline <- splinefun(G_l)
G_s_s <- G_s_spline(sb_bids_sub)
G_l_l <- G_l_spline(lb_bids_sub)
G_l_s_105 <- G_l_spline(1.05 * sb_bids_sub)
G_s_l_105 <- G_s_spline(lb_bids_sub / 1.05)

n_S <- 1
n_L <- 3
cost_small <- sb_bids_sub - 1 / (((n_S - 1) * g_s_s) / (1 - G_s_s)
   + (n_L * g_l_s_105) / (1 - G_l_s_105))
cost_large <- lb_bids_sub - 1 / ((n_S * g_s_l_105) / (1 -
   G_s_l_105) + ((n_L- 1) * g_l_l) / (1 - G_l_l))

png("./src/imgs/f_s.png")
plot(density(cost_small[cost_small > 0 & is.na(cost_small) ==
   FALSE]),
     main = "Estimated Distribution of Small Business Costs", xlim
        = c(0, 5.5e6))
dev.off()
png("./src/imgs/f_l.png")
plot(density(cost_large[cost_large > 0 & is.na(cost_large) ==
   FALSE]),
     main = "Estimated Prob Distribution of Large Business Costs",
        xlim = c(0, 5.5e6))
dev.off()

png("./src/imgs/F_l.png")
plot(ecdf(cost_large[cost_large > 0 & is.na(cost_large) == FALSE]),
     main = "Estimated CDF of Large Business Costs", xlim = c(0,
        5.5e6))
dev.off()
png("./src/imgs/F_s.png")
plot(ecdf(cost_small[cost_small > 0 & is.na(cost_small) == FALSE]),
     main = "Estimated CDF of Small Business Costs", xlim = c(0,
        5.5e6))
dev.off()
```

4 Conclusion

We were able to obtain estimates of the distribution of values for bidders on CalTrans procurement projects using data on their bids and estimates of the true cost. This is because there is an optimal bidding function which we assume the bidders followed. The work was complicated by the existence of small- and large-business type bidders.

5 Appendix - Code

The following is the R code used to prepare the data section of the paper.

```
load("./src/code/Caltrans_Data/caltransdata.RData")

# FIXME: take out auctions with only one bidder

# summary stats on the data from 705 auctions
# here, getting the number of types of bidders (either 1 or 2)
types_bidders <- vector(length =
   length(unique(caltransdata$ProjectID)))
auction_rows <- match(unique(caltransdata$ProjectID),
   caltransdata$ProjectID)
auction_indexer <- 1
for (auction in auction_rows) {
  types_bidders[auction_indexer] <-
    (caltransdata$NumberofSmallBusinessBidders[auction] != 0) +
    (caltransdata$NumberofLargeBusinessBidders[auction] != 0)
  auction_indexer <- auction_indexer + 1
}

sb_bids <- caltransdata[caltransdata$SmallBusinessPreference == 1,
   ]$Bid
num_bidders <- caltransdata$NumberofSmallBusinessBidders +
  caltransdata$NumberofLargeBusinessBidders

my_smry <- function(x) {
  return(c(mean(x), sd(x), min(x), max(x)))
}
all_bids_smry <- my_smry(caltransdata$Bid)
sb_bids_smry <- my_smry(sb_bids)
num_bidders_smry <- my_smry(num_bidders)
num_types_bidders_smry <- my_smry(types_bidders)
eng_est_smry <- my_smry(caltransdata$Estimate)
workdays_smry <- my_smry(caltransdata$WorkDays)
summary_stats <- matrix(Reduce(c, list(all_bids_smry, sb_bids_smry,
                                       num_bidders_smry,
                                       num_types_bidders_smry,
                                       eng_est_smry,
                                          workdays_smry)),
                        nrow = 6, byrow = TRUE)
colnames(summary_stats) <- c("Mean", "Standard Deviation",
   "Minimum", "Maximum")
rownames(summary_stats) <- c("Bids", "Small Business Bids", "Number
   of Bidders",
                             "Business types present",
                             "Engineer's Estimates", "Workdays")
library(xtable)
print(xtable(summary_stats), latex.environments = NULL, booktabs =
   TRUE,
      file = "./src/sections/data-summary.tex")

# how close are the (winning) bids to the engineer's estimate?
find_winning_bid <- function(x) {
  low_bidder <- which.min(x$Bid)
  low_bid <- min(x$Bid)
  if (x$SmallBusinessPreference[low_bidder] == 1 ||
      sum(x$NumberofSmallBusinessBidders) == 0) {
    return(low_bid)
  }
  sb <- x[x$SmallBusinessPreference == 1, ]$Bid
  low_sb_bid <- min(sb)
  if (low_sb_bid / low_bid < 1.05) {
    return(low_sb_bid)
  } else {
    return(low_bid)
  }
}
winning_bids <- by(caltransdata, factor(caltransdata$ProjectID),
                   find_winning_bid)
mean(winning_bids - caltransdata[auction_rows, ]$Estimate)

# pdfs and cdfs

pdf_sb <- density(sb_bids, bw = "UCV")
png("src/imgs/sb-pdf.png")
hist(sb_bids, breaks = 100, freq = FALSE, main = "Small Business
   Bids PDF",
     xlab = "Density", ylab = "Bids", xlim = c(0, 5e6), ylim = c(0,
        2e-6))
lines(pdf_sb, col = "firebrick")
dev.off()

lb_bids <- caltransdata[caltransdata$SmallBusinessPreference == 0,
   ]$Bid
lb_bw <- bw.ucv(lb_bids, lower = 1e-6, upper = 1e6)
pdf_lb <- density(lb_bids, bw = lb_bw)
png("./src/imgs/lb-pdf.png")
hist(lb_bids, breaks = 800, freq = FALSE, main = "Large Business
   Bids PDF",
     xlab = "Density", ylab = "Bids", xlim = c(0, 6e6), ylim = c(0,
        1.8e-6))
lines(pdf_sb, col = "firebrick")
dev.off()

log_lb_bids <- log(lb_bids)
png("./src/imgs/log-lb-pdf.png")
hist(log_lb_bids, breaks = 30, freq = FALSE,
     main = "Log of Large Business Bids PDF", xlab = "Density",
        ylab = "Bids",
     xlim = c(10, 20))
pdf_log_lb <- density(log_lb_bids)
lines(pdf_log_lb)
dev.off()

log_sb_bids <- log(sb_bids)
png("./src/imgs/log-sb-pdf.png")
hist(log_sb_bids, breaks = 30, freq = FALSE,
     main = "Log of Small Business Bids PDF", xlab = "ln(Bids)",
     ylab = "Density", xlim = c(10, 18))
pdf_log_sb <- density(log_sb_bids)
lines(pdf_log_sb)
dev.off()

cdf_lb_bids <- cumsum(lb_bids) / sum(lb_bids)
cdf_sb_bids <- cumsum(sb_bids) / sum(sb_bids)
```

```
png("./src/imgs/cdf.png")
# CDFs of the bids for small and large businesses
plot(ecdf(lb_bids), xlim = c(0, 8e6), col = "red", lty = 4,
     main = "CDF of Large and Small Business Bids", xlab = "bid")
lines(ecdf(sb_bids), col = "blue", lty = 3)
legend("bottomright", legend = c("Large", "Small"),
       col = c("red", "blue"), lty = c(4, 3))
dev.off()

# regressions
calt_full <- cbind(num_bidders, caltransdata)

calt_full <- calt_full[calt_full$NumberofSmallBusinessBidders +
                       calt_full$NumberofLargeBusinessBidders != 1,
                          ]

full_reg <- lm(Bid ~ num_bidders + Estimate + WorkDays, data =
   calt_full)
print(xtable(summary(full_reg)), latex.environments = NULL,
   booktabs = TRUE,
      file = "./src/sections/data-regressions.tex")

sb_reg <- lm(Bid ~ NumberofSmallBusinessBidders + Estimate +
   WorkDays,
             data =
                caltransdata[caltransdata$SmallBusinessPreference
                == 1, ])
print(xtable(summary(full_reg)), latex.environments = NULL,
   booktabs = TRUE,
      file = "./src/sections/data-regressions.tex", append = TRUE)


lb_reg <- lm(Bid ~ NumberofLargeBusinessBidders + Estimate +
   WorkDays,
             data =
                caltransdata[caltransdata$SmallBusinessPreference
                == 0, ])
print(xtable(summary(full_reg)), latex.environments = NULL,
   booktabs = TRUE,
      file = "./src/sections/data-regressions.tex", append = TRUE)



# Estimating section: want picture of denominator terms in slide
   10, lecture ?
# density/cdf for type 1/type 2: 8 total
# small large: pdf/cdf cost, pdf/cdf bid (f, g, F, G)
# we are estimating the whole term from that slide: the COST
# evaluate density of type 1 bids at the type 2 bids value -
   ksdensity in matlab
# if you get a negative/infinity, check if g_1/2 is small - likely
```

```
   numerical
# calculation error
# all these pictures are for the median engineer's estimate
# the c we get is automatically conditional on that because right
   side is

# x -> (Cost) F_(L, S)(dot | x) -> (Bids) G_(L, S)(dot | X, n_L,
   n_S)

# we are assuming n does not affect f. in real world, does:
   decision to enter.
# larger n means bid more aggressively, # so affects g.
# select subset of auctions with the representative combination of
   n_L or n_S
# eg how many had 2 n_L and 5 n_S is largest percent, take those
# alternatively, not doing this requires a bit more work. x/bids
   are continuous,
# so we have to mix integers (n_S, n_L) in

# then, conditioning on x, bayes's rule
# so how do we get a joint probability density of bids and x?
# estimate marginal of X, divide first by second, integrate

# n by 2 matrix of n_S and n_L for each i in n
dt_bids <- cbind(calt_full$NumberofSmallBusinessBidders,
                 calt_full$NumberofLargeBusinessBidders)
rows <- c()
for (row_i in seq_len(nrow(dt_bids))) {
  current_row <- dt_bids[row_i, ]
  row_as_char <- paste(current_row, collapse = " ")
  rows <- c(rows, row_as_char)
}
sort(table(rows), decreasing = TRUE)
calt_subset <- calt_full[calt_full$NumberofSmallBusinessBidders ==
   1 &
                         calt_full$NumberofLargeBusinessBidders ==
                            3, ]

# vectors of large/small bids from our subset
sb_bids_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   1, ]$Bid
lb_bids_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   0, ]$Bid
# change these further down
ests_subset <- calt_subset$Estimate
# there is just one estimate per auction, so median estimate is
   same for large/small
median_estimate <- median(ests_subset)
med_est_vec <- rep(median_estimate, length(sb_bids_sub)) # and lb
   bids too
```

```
sb_ests_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   1, ]$Estimate
# this is just sb_ests_sub with each entry repeated three times
lb_ests_sub <- calt_subset[calt_subset$SmallBusinessPreference ==
   0, ]$Estimate

# repeat the median appropriate number of times
sb_med_est_vec <- rep(median_estimate, length(sb_bids_sub))
lb_med_est_vec <- rep(median_estimate, length(lb_bids_sub))


library(hdrcde)
library(devtools)
# FIXME: there's no point using nix if I do this :(
devtools::install_github("https://github.com/sethmcg/climod")
library(climod)

# cde(x,y) gives p(y|x)
ests_grid <- seq(from = min(sb_ests_sub), to = max(sb_ests_sub),
   length = 21)
g_l <- cde(lb_ests_sub, lb_bids_sub, deg = 1, link = "log",
   nxmargin = 21,
          x.name = "Estimates", y.name = "Large Business Bids")
png("./src/imgs/g_l_cond.png")
plot(g_l)
dev.off()
png("./src/imgs/g_l_median.png")
plot(g_l$y, g_l$z[3, ], main = "Large Bid PDF",
     xlab = "Bid", ylab = "Estimated Density") # at the median
        estimate
dev.off()
g_s <- cde(sb_ests_sub, sb_bids_sub, deg = 1, link = "log",
   nxmargin = 21,
          x.name = "Estimates", y.name = "Small Business Bids")
png("./src/imgs/g_s_cond.png")
plot(g_s)
dev.off()
png("./src/imgs/g_s_median.png")
plot(g_s$y, g_s$z[3, ], main = "Small Bid PDF",
     xlab = "Bid", ylab = "Estimated Density")
dev.off()

# I think the issue I'm running into is that we are getting a joint
   PDF
# which integrates to one over the entire space of estimates. but
   we are just taking
# a subset of the median estimate. so it doesn't integrate to one.
# but, pdf2cdf can normalize to 1, so it's ok.

# y is a grid of length 100 plugged in for bids
# z has 100 columns: the third row is those evaluated at the median
```

```
   estimate
G_l <- pdf2cdf(g_l$z[3, ], g_l$y)
png("./src/imgs/G_l.png")
plot(G_l, main = "Large Bid CDF",
     xlab = "Bid", ylab = "Estimated Cumulative Density")
dev.off()
G_s <- pdf2cdf(g_s$z[3, ], g_s$y)
png("./src/imgs/G_s.png")
plot(G_s, main = "Small Bid CDF",
     xlab = "Bid", ylab = "Estimated Cumulative Density")
dev.off()

g_s_spline <- splinefun(g_s$y, g_s$z[3, ])
# integrate(g_s_spline, 100, 6e6) gives 0.996
g_l_spline <- splinefun(g_l$y, g_l$z[3, ])
g_s_s <- g_s_spline(sb_bids_sub)
g_l_l <- g_l_spline(lb_bids_sub)
g_l_s_105 <- g_l_spline(1.05 * sb_bids_sub)
g_s_l_105 <- g_s_spline(lb_bids_sub / 1.05)

G_s_spline <- splinefun(G_s)
G_l_spline <- splinefun(G_l)
G_s_s <- G_s_spline(sb_bids_sub)
G_l_l <- G_l_spline(lb_bids_sub)
G_l_s_105 <- G_l_spline(1.05 * sb_bids_sub)
G_s_l_105 <- G_s_spline(lb_bids_sub / 1.05)

n_S <- 1
n_L <- 3
cost_small <- sb_bids_sub - 1 / (((n_S - 1) * g_s_s) / (1 - G_s_s)
   + (n_L * g_l_s_105) / (1 - G_l_s_105))
cost_large <- lb_bids_sub - 1 / ((n_S * g_s_l_105) / (1 -
   G_s_l_105) + ((n_L- 1) * g_l_l) / (1 - G_l_l))

png("./src/imgs/f_s.png")
plot(density(cost_small[cost_small > 0 & is.na(cost_small) ==
   FALSE]),
     main = "Estimated Distribution of Small Business Costs", xlim
        = c(0, 5.5e6))
dev.off()
png("./src/imgs/f_l.png")
plot(density(cost_large[cost_large > 0 & is.na(cost_large) ==
   FALSE]),
     main = "Estimated Prob Distribution of Large Business Costs",
        xlim = c(0, 5.5e6))
dev.off()

png("./src/imgs/F_l.png")
plot(ecdf(cost_large[cost_large > 0 & is.na(cost_large) == FALSE]),
     main = "Estimated CDF of Large Business Costs", xlim = c(0,
        5.5e6))
```

```
dev.off()
png("./src/imgs/F_s.png")
plot(ecdf(cost_small[cost_small > 0 & is.na(cost_small) == FALSE]),
     main = "Estimated CDF of Small Business Costs", xlim = c(0,
        5.5e6))
dev.off()

# these do not have 5 percent rule
# cost_small <- sb_bids_sub - 1 /
#   ( ((n_S - 1) * g_s_s) / (1 - G_s_s) + (n_L * g_l_s) / (1 -
   G_l_s) )
#
# cost_large <- lb_bids_sub - 1 /
#   ( (n_S * g_s_l) / (1 - G_s_l) + ((n_L - 1) * g_l_l) / (1 -
   G_l_l) )

##### START UNUSED CODE ######

library(ks) # density does not support higher dimensional kde
# step 3 conditional bids evaluated at bids (not b)
# also returns a vector of probabilities
conditional_bids <- function(bids, estimates, to_pred) {
  # step 1 joint PDF of bid and estimates evaluated at BID and
     median
  # trains on some bids/estimates, returns a vector of estimates
     for other bids (and median est)
  kde_joint_pdf <- ks::kde(x = cbind(bids, estimates),
                           eval.points = to_pred)
  joint_bids_est <- kde_joint_pdf$estimate

  # step 2 marginal of estimates evaluated at its median
  # returns a scalar
  marg_kde_pdf <- density(estimates, bw = 0.01) # look into bw issue
  myspline <- approxfun(marg_kde_pdf)
  marginal_bids <- myspline(median(estimates))

  return(joint_bids_est / marginal_bids)
}

# get functions for our g's so we can get G's by numerical
   integration
vgrid <- seq(1e3, 5.9e6, by = 500)
vgrid_medians <- rep(median_estimate, length(vgrid))
# create a function for g_l and for g_s
# I think splinefun is better than approxfun?
gl_grid <- cbind(vgrid, conditional_bids(lb_bids_sub, lb_ests_sub,
   cbind(vgrid, vgrid_medians)))
gl_spline <- splinefun(gl_grid)
gs_grid <- cbind(vgrid, conditional_bids(sb_bids_sub, sb_ests_sub,
   cbind(vgrid, vgrid_medians)))
gs_spline <- splinefun(gs_grid)
```

```
G_s_f <- function(evalpoint) {
  out <- integrate(gs_spline, 1e3, evalpoint)$value
  return(out)
}

G_l_f <- function(evalpoint) {
  out <- integrate(gl_spline, 1e3, evalpoint)$value
  return(out)
}

# lines - sort and then rownames <- NULL
# plot(sb_bids_sub, g_s_s)
plot(gs_grid)
plot(lb_bids_sub, g_l_l)
# TODO make smooth lines
plot(sb_bids_sub, G_s_s)
plot(lb_bids_sub, G_l_l)

# evaluate the conditional densities for bids
# g_s_l is g_s(large bids)
# evaluate at the bids because these are for plugging into the cost
   function
g_s_s <- conditional_bids(sb_bids_sub, sb_ests_sub,
   cbind(sb_bids_sub, sb_med_est_vec))
g_l_l <- conditional_bids(lb_bids_sub, lb_ests_sub,
   cbind(lb_bids_sub, lb_med_est_vec))
g_l_s_105 <- conditional_bids(lb_bids_sub, lb_ests_sub, cbind(1.05
   * sb_bids_sub, sb_med_est_vec))
g_s_l_105 <- conditional_bids(sb_bids_sub, sb_ests_sub,
   cbind(lb_bids_sub / 1.05, lb_med_est_vec))

G_s_s <- sapply(sb_bids_sub, G_s_f)
G_l_l <- sapply(lb_bids_sub, G_l_f)
# same thing - CDFs with 1.05 * bids
G_l_s_105 <- sapply(1.05 * sb_bids_sub, G_l_f)
G_s_l_105 <- sapply(lb_bids_sub / 1.05, G_s_f)

##### END UNUSED CODE #####
```