Skills Test

Brad Chattergoon

Data Task

Load Data

We display a brief look at the dataset for reference.

```
df = read.dta13('caltrans-bidders1999-2008.dta')
glimpse(df)
```

```
#> Observations: 7,473
#> Variables: 91
#> $ c                      <int> 830, 897, 551, 1268, 231, 231, 231, 1293, 43...
#> $ contract               <chr> "06-397004", "06-470204", "04-2R0804", "09-2...
#> $ locationcode           <chr> "06-MAD-99-13.0/23.0", "06-FRE-180-61.4/64.1...
#> $ location               <chr> "IN MADERA COUNTY NEAR MADERA FROM 0.4 KM NO...
#> $ biddate                <chr> "4/9/2002", "10/30/2002", "6/5/2002", "11/6/...
#> $ BidDate                <date> 2002-04-09, 2002-10-30, 2002-06-05, 2002-11...
#> $ job                    <chr> "ASPHALT CONCRETE OVERLAY", "RESURFACE EXIST...
#> $ district               <chr> "06", "06", "04", "09", "02", "02", "02", "0...
#> $ nbidders               <int> 4, 10, 7, 7, 5, 5, 5, 4, 3, 4, 3, 7, 7, 3, 6...
#> $ engestimate            <dbl> 7815000, 745000, 1470000, 3688000, 166000, 1...
#> $ bidderid               <int> 135, 149, 99, 149, 54, 213, 125, 22, 99, 473...
#> $ bidder                 <chr> "KIEWIT PACIFIC CO", "M J MENEFEE CONST INC"...
#> $ address                <chr> "P O BOX 1769 VANCOUVER WA  98668", "P O BOX...
#> $ bidrank                <int> 2, 1, 7, 7, 1, 2, 4, 1, 2, 1, 1, 7, 4, 1, 6,...
#> $ bidtotal               <dbl> 6560014, 539116, 1352711, 2672731, 184817, 2...
#> $ winner                 <int> 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0,...
#> $ winning_bid            <dbl> 5791305, 539116, 1129751, 2015102, 184817, 1...
#> $ re                     <chr> "<BLANK>", "<BLANK>", "<BLANK>", "<BLANK>", ...
#> $ fedfund                <int> 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 1, 1, 0,...
#> $ funddes1               <chr> "ACNH-P099-(443)E", "", "", "ACNH-P006-(008)...
#> $ funddes2               <chr> "", "", "", "", "", "", "", "", "AC-P113-(02...
#> $ funddes3               <chr> "", "", "", "", "", "", "", "", "", "", "", ...
#> $ numberofcontractitems  <int> 44, 14, 20, 28, 9, 9, 9, 18, 40, 23, 17, 56,...
#> $ adjustments            <dbl> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ posAdj                 <dbl> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ negAdj                 <dbl> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ extrawork              <dbl> NA, NA, 32656, NA, NA, NA, NA, NA, NA, NA, N...
#> $ deductions             <dbl> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ bidtotal_actual        <dbl> 0, 0, 1234114, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
#> $ liquidateddamages      <chr> "", "", "", "", "", "", "", "", "", "", "", ...
#> $ PCT                    <dbl> NA, NA, -0.10166, NA, NA, NA, NA, NA, NA, NA...
#> $ sum_ccdbover           <dbl> 0, 0, -68800, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
#> $ CCestprojsize          <dbl> 5547885, 718624, 1132831, 2202587, 252152, 2...
#> $ CCactprojsize          <dbl> 0, 0, 1045202, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
```

```
#> $ distance                <dbl> 733.0, 9.5, 91.2, 239.0, 168.0, 202.0, 124.0...
#> $ time                    <dbl> 689, 16, 114, 293, 182, 217, 188, 76, 94, 90...
#> $ rivaltime               <dbl> 41, 4, 45, 185, 188, 182, 182, 145, 29, 42, ...
#> $ backlog                 <dbl> 0, 0, 0, 1156655, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
#> $ contractdays            <int> NA, NA, 35, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ workingdays             <int> NA, NA, 28, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ dateapproved            <chr> "", "", "7/12/2002", "", "", "", "", "", "",...
#> $ datebeginconstr         <chr> "", "", "9/10/2002", "", "", "", "", "", "",...
#> $ dateworkstarted         <chr> "", "", "9/3/2002", "", "", "", "", "", "", ...
#> $ datecompleted           <chr> "", "", "11/6/2002", "", "", "", "", "", "",...
#> $ weatherdays             <int> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ ccodays                 <int> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ otherdays               <int> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ percentcompleted        <chr> "", "", "100%", "", "", "", "", "", "", "", ...
#> $ percentelapsed          <chr> "", "", "100%", "", "", "", "", "", "", "", ...
#> $ contractdays_final      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ badcontract             <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ rebid_dum               <int> NA, NA, NA, NA, NA, NA, NA, 1, 1, NA, NA, NA...
#> $ aplusb                  <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ missingitemdata         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
#> $ altered                 <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ irregular               <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
#> $ irrContract             <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ noFP                    <int> 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
#> $ noFP2000                <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
#> $ finaldum                <int> NA, NA, 1, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ exclude                 <int> 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
#> $ BidYear                 <int> 2002, 2002, 2002, 2002, 2002, 2002, 2002, 20...
#> $ BidMonth                <int> 4, 10, 6, 11, 3, 3, 3, 6, 9, 3, 6, 3, 11, 9,...
#> $ winner_id               <int> 262, 149, 48, 104, 54, 54, 54, 22, 104, 473,...
#> $ bidrank_off             <int> 2, 1, 7, 7, 1, 2, 4, 1, 2, 1, 1, 7, 4, 1, 6,...
#> $ timerank                <int> 4, 4, 7, 3, 1, 4, 2, 1, 2, 4, 1, 3, 6, 2, 5,...
#> $ distrank                <int> 4, 4, 7, 3, 3, 4, 2, 1, 2, 4, 2, 3, 6, 2, 5,...
#> $ dist1                   <dbl> 40.7, 1.9, 26.9, 163.0, 124.0, 124.0, 124.0,...
#> $ dist2                   <dbl> 110.0, 5.9, 33.1, 214.0, 124.0, 124.0, 124.0...
#> $ bidderlat               <dbl> 45.64, 36.63, 38.39, 36.63, 39.02, 38.56, 40...
#> $ bidderlong              <dbl> -122.6, -119.7, -122.7, -119.7, -121.1, -121...
#> $ contractlat             <dbl> 37.04, 36.74, 37.28, 37.45, 40.28, 40.28, 40...
#> $ contractlong            <dbl> -120.1, -119.7, -122.4, -118.3, -120.5, -120...
#> $ marketsharepct          <dbl> 2.560787, 0.088859, 0.915426, 0.088859, 0.00...
#> $ time1                   <dbl> 41, 4, 45, 185, 182, 182, 182, 76, 29, 42, 9...
#> $ time2                   <dbl> 135, 13, 48, 254, 188, 188, 188, 145, 94, 63...
#> $ StartDate               <date> NA, NA, 2002-09-03, NA, NA, NA, NA, NA, NA,...
#> $ sum_itemover            <dbl> 0, 0, -104999, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
#> $ bsavail                 <chr> "1", "1", "1", "1", "1", "1", "1", "1", "1",...
#> $ daysoverrun             <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ cdayworkstarted         <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ expectedcompletion      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ expectedcompletiondate <chr> "", "", "", "", "", "", "", "", "", "", "", ...
#> $ calendardayslate        <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ totaldays               <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
#> $ total_won               <dbl> 0, 539116, 0, 0, 184817, 0, 0, 1166666, 0, 3...
#> $ total_paid              <dbl> NA, NA, 0, NA, NA, NA, NA, NA, NA, NA, NA, N...
#> $ utilrank                <int> 1, 1, 5, 7, 3, 5, 1, 1, 1, 3, 1, 5, 3, 2, 5,...
```

```
#> $ util1                  <dbl> 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.00...
#> $ util2                  <dbl> 0.00000, 0.00000, 0.00000, 0.00000, 0.00000,...
#> $ re_count               <int> 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, 69, ...
```

```
# contract is project code
num_projects = nrow(df %>% distinct(contract))
```

Question 1

We can see that the number of projects is 1584. We would want to verify that this makes sense by noting that the projects are numbered from 1 so we can double check this number and look for data drop-out by checking that indeed the last project number is 1584.

```
tail(df %>% distinct(c) %>% arrange(c))
```

```
#>         c
#> 1579 1579
#> 1580 1580
#> 1581 1581
#> 1582 1582
#> 1583 1583
#> 1584 1584
```

Question 2

```
# Extract start year
df = df %>% mutate(start_year = year(mdy(dateworkstarted)))
# Extract second lowest bid
second_lowest_bid = df %>%
  group_by(contract) %>%
  select(contract,bidtotal) %>%
  filter(bidtotal != min(bidtotal)) %>%
  slice(which.min(bidtotal)) %>%
  rename('secondbid' = 'bidtotal')
df = df %>%
  left_join(second_lowest_bid, by = 'contract')
# Filter exclusions and created new variables
df = df %>%
  filter(exclude == 0, start_year < 2006) %>%
  # Note here that I define normbid based on the normalized total bid for use later in the
  # regression analysis, but then filter on winners to get the project level summary stats
  mutate(
    normbid = bidtotal/engestimate,
    markup = (bidtotal - engestimate)/engestimate,
    moneyontable = winning_bid - secondbid,
    normmoney = moneyontable/engestimate
  )

projects = df %>%
  filter(winner  == 1) %>%
  select(
    contract,
    winning_bid,
    secondbid,
    normbid,
```

```
    markup,
    moneyontable,
    normmoney,
    nbidders
  )

stargazer(projects %>% select(-contract),
          omit.summary.stat = c("p25", "p75"),
          header = FALSE,
          title = 'Summary Statistics for Projects in Sample',
          digits = 2)
```

Table 1: Summary Statistics for Projects in Sample

Statistic	N	Mean	St. Dev.	Min	Max
winning_bid	776	2,704,962.00	7,031,290.00	51,626.00	99,599,232.00
secondbid	776	2,884,106.00	7,454,081.00	60,764	106,988,376
normbid	776	0.93	0.18	0.38	1.82
markup	776	−0.07	0.18	−0.62	0.82
moneyontable	776	−179,144.00	482,491.00	−7,389,144.00	−67.50
normmoney	776	−0.07	0.07	−0.75	−0.0002
nbidders	776	4.52	2.15	2	19

There are just a little under half of the total number of projects left in our sample (half = 792). In asking how the summary table would be different if we had kept them I suppose you want me to intuit a description rather than simply print the summary table without the exclusions. The winnind_bid mean would probably be higher since we exlcude projects where the winning was not the lowest bid, normbid would be higher as well due to this. The markup average might be higher due to this as well, but since this based on the difference between the winning_bid and engestimate it is possible that the engestimates for these otherwise excluded projects could have been higher than the winning_bid which would put downward pressure on the markup average. The secondbid average is hard to assess since some of these excluded projects only had one bid. Consequently moneyontable and normmoney would be difficult to assess. nbidders would surely be lower in average since we remove projects with only one bidder.

Question 3

```
df = df %>%
  group_by(bidderid) %>%
  mutate(capacity = max(backlog)) %>%
  ungroup() %>%
  mutate(util = ifelse(backlog == 0, 0, backlog/capacity)) %>%
  group_by(contract) %>%
  mutate(distank = rank(distance, ties.method = 'first'),
         utilrank = rank(util, ties.method = 'first')) %>%
  mutate(rivaldist = ifelse(distrank == 1, distance[which(distrank == 2)], min(distance)),
         rivalutil = ifelse(utilrank == 1, util[which(utilrank == 2)], min(util))) %>%
  ungroup()

stargazer(as.data.frame(df %>% select(distance, rivaldist, backlog, capacity, util, rivalutil)),
          omit.summary.stat = c("p25", "p75"),
          header = FALSE,
```

```
          title = 'Summary Statistics for Utilization and Distance',
          digits = 1)
```

Table 2: Summary Statistics for Utilization and Distance

Statistic	N	Mean	St. Dev.	Min	Max
distance	3,504	94.0	130.6	0.1	2,857.0
rivaldist	3,504	37.7	52.0	0.1	618.6
backlog	3,504	4,685,233.0	14,502,303.0	0	150,411,535
capacity	3,504	25,718,850.0	50,327,847.0	0.0	150,411,535.0
util	3,504	0.1	0.3	0	1
rivalutil	3,504	0.02	0.1	0	1

For the utility calculation I had to make a decision on how to handle bidders with a zero capacity (due to 0 max backlog) which are likely to not have prior backlog in their records for whatever reason. This could be some kind of error but for this exercise I'll accept the data as correct and assume that if backlog is 0 then independent of capacity the utilization is set to 0.

Question 4

```
df = df %>%
  mutate(caltrans_total_paid = sum(total_paid, na.rm = TRUE)) %>%
  group_by(bidderid) %>%
  mutate(lifetime_paid = sum(total_paid)) %>%
  ungroup() %>%
  mutate(fringe = ifelse(lifetime_paid/caltrans_total_paid < 0.01, 1, 0))

comparison = df %>%
  select(total_won, total_paid, winner, bidderid) %>%
  mutate(caltrans_total_paid = sum(total_paid, na.rm = TRUE),
         caltrans_total_won = sum(total_won, na.rm = TRUE),
         caltrans_auctions = sum(winner, na.rm = TRUE)) %>%
  group_by(bidderid) %>%
  mutate(lifetime_paid = sum(total_paid),
         lifetime_won = sum(total_won),
         lifetime_auctions = sum(winner)) %>%
  ungroup() %>%
  mutate(fringe_paid = ifelse(lifetime_paid/caltrans_total_paid < 0.01, 1, 0),
         fringe_won = ifelse(lifetime_won/caltrans_total_won < 0.01, 1, 0),
         fringe_auctions = ifelse(lifetime_auctions/caltrans_auctions < 0.01, 1, 0)) %>%
  distinct(bidderid, .keep_all = TRUE) %>%
  summarise('Total Fringe Paid' = sum(fringe_paid),
            'Total Fringe Won' = sum(fringe_won),
            'Total Fringe Auctions' = sum(fringe_auctions)) %>%
  distinct(.keep_all = TRUE)


stargazer(as.data.frame(comparison),
          header = FALSE,
          title = 'Comparison of Fringe Classifications',
          summary = FALSE,
```

```
          rownames = FALSE,
          table.placement = 'h')
```

Table 3: Comparison of Fringe Classifications

Total Fringe Paid	Total Fringe Won	Total Fringe Auctions
325	325	318

There are comparable numbers of fringe bidders for Total_Paid and Total_Won but much less fringe bidders for the metric determined by the number of auctions won. This is likely due to some projects being very large and some projects being smaller. Smaller projects would have a smaller pay out but smaller firms would be able to bid on them whereas smaller firms might not have the ability to handle the much larger projects (which would also pay more) and don't bid on them. If we use the paid or won metric we will bias towards labelling smaller firms as fringe relative to the auctions won metric.

Question 5

If we have that the total_won metric leads to 10 fewer fringe bidders than the total_paid metric, we are equivalently saying that that there are 10 more bidders who fall into the 'at least 1% of all winnings awarded by Caltrans' bucket than the equivalent for the total_paid metric. In other words, the total_paid metric leads to more bidders classified as fringe than total_won. Given that the paid metric is the amount paid out and is different from the total amount awarded due to to "project adjustments made between the auction and whenever the project is completed", then we need to understand the set of actions that make up "project adjustments". Let's make some assumptions, the first is that bidders make a profit on each unit of component sold to Caltrains. The second is that project adjustments do not allow for price changes and only allow for changes in component amounts. Then what could explain the increase in fringe bidders in the total_paid metric is that certain companies strategically bid to win auctions where they believe Caltrans has underestimated the amount of a component needed to finish a project. This would allow these strategic companies to bid below other bidders and win the contract but still get the same expected profits. $\pi = (p - c) * q$, and so these strategic companies may be planning to have the same π as other companies but their price is lower and quantity is higher. This strategy would scale to make total_paid a larger number than total_won, and companies who do not employ this strategy would be more likely to be classified as fringe in the total_paid metric than in the total_won metric.\

I'm not quite sure if this would be a problem because I don't quite understand what the context of the relationship between fringe status and strategic bidding that we would be interpreting is. If the context is that we are trying to investigate whether there is this type of strategic bidding at play then this is exactly what we want to see to help us test it. If we want to investigate something else that would be affected by this strategic bidding then this would be a problem.

Questions 6-9

```
df = df %>%
  mutate(distance_reg = distance/100,
         rivaldist_reg = rivaldist/100)

model_q6 = df %>% lm(normbid ~ distance_reg +
                               rivaldist_reg, data = .)
```

```
model_q7 = df %>% lm(normbid ~ distance_reg +
                         rivaldist_reg +
                         util +
```

```
                        rivalutil +
                        fringe +
                        nbidders, data = .)
```

```
model_q8 = df %>% lm(normbid ~ distance_reg +
                        rivaldist_reg +
                        util +
                        rivalutil +
                        fringe +
                        nbidders +
                        factor(bidderid), data = .)

model_q8_2 = df %>% lm(normbid ~ distance_reg +
                        rivaldist_reg +
                        util +
                        rivalutil +
                        fringe +
                        nbidders +
                        factor(contract), data = .)
```

```
df = df %>%
  mutate(normbid_actual = bidtotal_actual/engestimate)

model_q9 = df %>% lm(normbid_actual ~ distance_reg +
                        rivaldist_reg +
                        util +
                        rivalutil +
                        fringe +
                        nbidders +
                        factor(bidderid), data = .)

model_q9_2 =  df %>% lm(normbid_actual ~ distance_reg +
                        rivaldist_reg +
                        util +
                        rivalutil +
                        fringe +
                        nbidders +
                        factor(contract), data = .)
```

```
stargazer(model_q9,
          model_q9_2,
          model_q6,
          model_q7,
          model_q8,
          model_q8_2,
          title = 'Regression Results',
          omit.stat = c('f','ser'),
          omit = c('bidderid','contract'),
          omit.labels = c("Bidder Fixed Effects","Conract Fixed Effects"),
          header = FALSE)
```

In using the fixed effects one of the effects must serve as the "zero" for the other effects to be taken relative to. I am not familiar with standard error clustering. I read a write up[1] about the paper on standard error

[1] https://blogs.worldbank.org/impactevaluations/when-should-you-cluster-standard-errors-new-wisdom-econometrics-

Table 4: Regression Results

	Dependent variable:					
	normbid_actual			normbid		
	(1)	(2)	(3)	(4)	(5)	(6)
distance_reg	0.007	0.007***	0.006*	0.008***	0.001	0.008***
	(0.005)	(0.003)	(0.003)	(0.003)	(0.004)	(0.002)
rivaldist_reg	0.016	−0.003	0.015**	0.005	0.012	−0.004
	(0.010)	(0.010)	(0.008)	(0.008)	(0.008)	(0.009)
util	0.037*	0.005		0.008	0.037**	0.002
	(0.019)	(0.012)		(0.016)	(0.016)	(0.012)
rivalutil	−0.100*	0.014		−0.099**	−0.091**	0.009
	(0.052)	(0.055)		(0.045)	(0.043)	(0.053)
fringe	−0.141	0.031***		0.045***	−0.028	0.035***
	(0.366)	(0.006)		(0.008)	(0.304)	(0.006)
nbidders	−0.008***	−0.269**		−0.013***	−0.015***	−0.166
	(0.002)	(0.134)		(0.002)	(0.002)	(0.128)
Constant	1.197***	1.609***	1.028***	1.078***	1.204***	1.319***
	(0.259)	(0.354)	(0.005)	(0.012)	(0.215)	(0.338)
Bidder Fixed Effects	Yes	No	No	No	Yes	No
Conract Fixed Effects	No	Yes	No	No	No	Yes
Observations	3,504	3,504	3,504	3,504	3,504	3,504
R^2	0.197	0.780	0.003	0.027	0.215	0.715
Adjusted R^2	0.108	0.717	0.002	0.025	0.128	0.634
Note:						*p<0.1; **p<0.05; ***p<0.01

clustering by Athey, Abadie, Imbens and Wooldridge, and based on the write up, we neither have a situation where the sample was selected in such a way that clustering is needed nor do we have a situation with a treatment effect that would require clustering. I may be wrong about this.

Starting with the Question 6 Model: The (Adj) R^2 on this is very poor, only 0.002, and I would not put any weight on this model but it seems like rival distance might be more important than firm distance in such a way that the larger the rival distance the larger the normbid.

Question 7 Model: (Adj) R^2 improves to 0.025 but this is still fairly low. We now find statistical significance on the distance, fringe status, rival utilization, and nbidders. The nbidders makes clear sense since we expect competition to decrease prices and this is indeed what we see. Fringe status correlates with a higher normbid which again makes sense since we have defined fringe to be such that they don't win/get paid much money by Caltrans and given that the lowest bid wins, fringe bidders must be bidding higher on average by construction. Distance being positive makes sense if firms are pricing in their delivery costs to their bid. The statistical insignificance of rival distance here is of note. This is suggestive that in this model firms do not adjust their bid based on their rivals' distance, but this should be taken into consideration with the statistically significant and negative coefficient on the rival utilization. It seems that firms might be reducing their bid when rival utilization is high but this doesn't make much sense strategically since high rival utilization should mean a higher normbid not lower. The rationale for this higher normbid would be that since rivals have high utilization, they might have to use more expensive resources to fulfill a bid, and so the appropriate bid strategy would be to have a higher normbid in response to high rival utilization. This departure from strategic intution combined with the low R^2 makes me heavily discount the validity of this model.

Question 8 Model 1: R^2 improves again to 0.128, which is better but still not very high. We use Bidder Fixed Effects for this model and we find statistical significance on bidder utilization, rival bidder utilization, and nbidders. The rival bidder utilization coefficient is again negative which as before does not seem in line with what we would expect. The utilization coefficient is positive which might be explained by firms needing to utilize more expensive resources when bidding on contracts when their utilization is high. The lack of statistical significance on fringe and the sign is concerning however. As described before, we choose fringe by construction to have a positive coefficient and we should expect some statistical significance given our construction. We might expect the R^2 on this model to be higher if we did find that some bidders are more strategic than others and therefore the normbid is tied strongly to certain bidders (since it is normalized we can account for differences across projects). The low R^2 value here is suggestive that there isn't a significant strategic input from bidders. All in all, this model doesn't seem to be the right one in fitting the data.

Question 8 Model 2: R^2 is quite high for this model at 0.634. We see statistical significance on the fringe coefficient and in the direction we should expect which is a good initial sign as well. The only other statistically significant coefficient is on the distance from the project location which again might be explained by firms pricing the distance into their bids for transportation costs if those are borne by the firm. We use fixed effects at the project level for this model. Our normbid is a function of the bidtotal normalized by the engestimate. The high R^2 for this model might be representative of the project level fixed effect capturing systematic differences between the engestimate and the true cost of the project that the bidders understand better. Given the high R^2 and the statistically significant coefficients, it seems there is no evidence of strategic bidding behaviors by the firms.

Question 9 Model 1: R^2 for this model is even lower than the R^2 for the corresponding Question 8 Model. Given that the bidtotal_actual represents the true value of the cost of the project, we might expect this regressand to be better able to capture the behavior of bidders in the model. That said, we run into the same problems with this model that we do in the corresponding Question 8 Model, and the R^2 is even lower suggesting that the model is not accurate.

Question 9 Model 2: R^2 for this model is even higher than the R^2 for the corresponding Question 8 Model. In the context of what we have said with regards to the normalized bidtotal_actual regressand, the increased R^2 gives evidence that this model might be closer to the true model of bidder behavior. We see the same statistically significant coefficients in the same directions but this time with the inclusion of significance

oracle

on nbidders, which we have argued before makes sense given the effects of competition. The statistically significant coefficients and directions combined with the high R^2 provide strong evidence that this market operates competitively and there isn't any meaningful strategic bidding practices among bidders that might adversely affect a competitive equilbrium.

Extension of Analysis

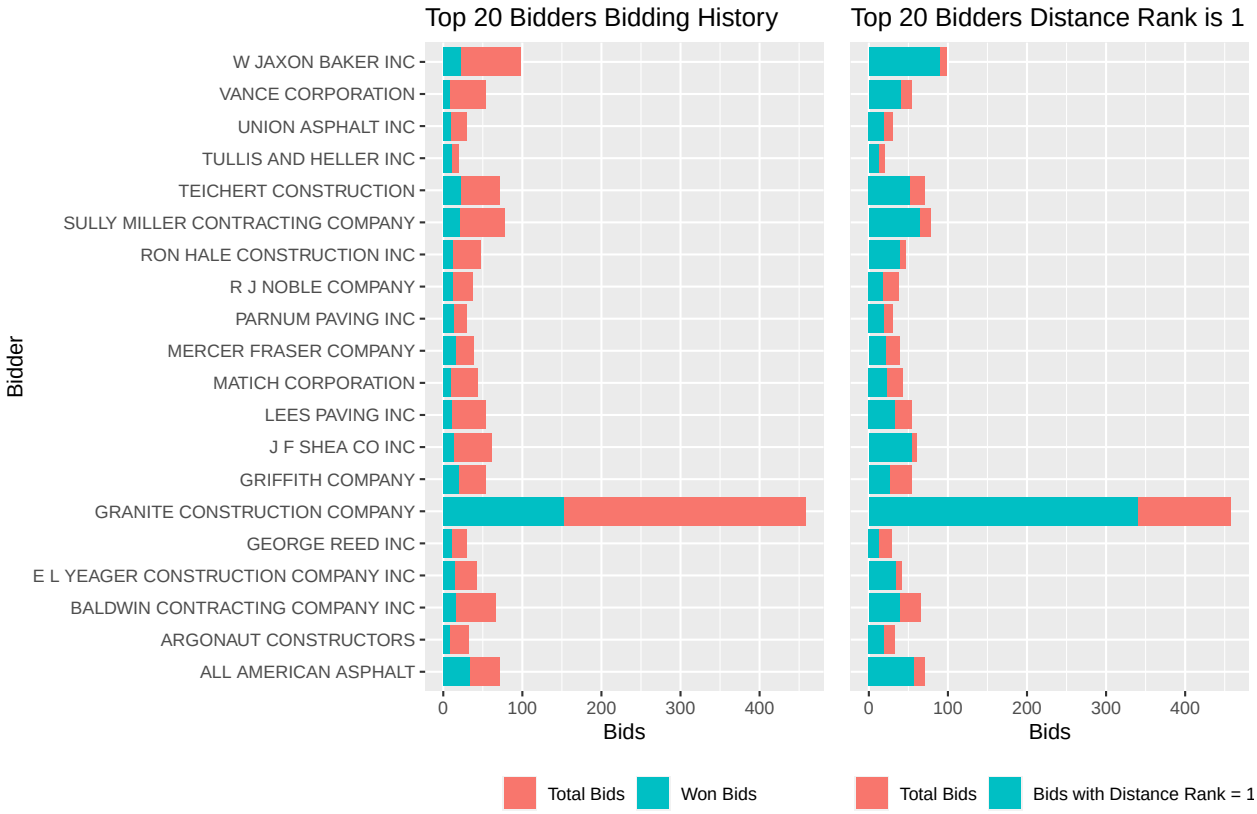
We extend the analysis briefly in two ways.

Extension 1

First, given that we find evidence for competitive bidding in the auction environment, we look at the bidding behavior of the top 20 most frequent bidders and their bid win rates when they are the lowest distance from the site (distance rank 1).

```
df = df %>%
  group_by(bidderid) %>%
  mutate(capacity = max(backlog)) %>%
  ungroup() %>%
  mutate(util = ifelse(backlog == 0, 0, backlog/capacity)) %>%
  group_by(contract) %>%
  mutate(distrank = rank(distance, ties.method = 'first'),
         utilrank = rank(util, ties.method = 'first')) %>%
  ungroup() %>%
  mutate(dist_rank_1 = ifelse(distrank == 1, 1, 0),
         util_rank_1 = ifelse(utilrank == 1, 1, 0))

bidders = df %>%
  group_by(bidder, bidderid) %>%
  summarise(n = n(), wins = sum(winner), avg_distance = mean(distance), distrank_1 = sum(dist_rank_1)) %
  arrange(desc(wins),desc(n)) %>%
  mutate(total = n - wins, total_distrank = n - distrank_1)

bidders_plot = bidders %>%
  pivot_longer(c(wins, total), names_to = 'type', values_to = 'vals')%>%
  head(40) %>%
  ggplot(aes(x = bidder, y = vals, fill = type)) +
  geom_bar(position = 'stack', stat = 'identity') +
  labs(title = 'Top 20 Bidders Bidding History') +
  xlab('Bidder') +
  ylab('Bids') +
  scale_fill_discrete(name = "", labels = c("Total Bids", "Won Bids")) +
  theme(legend.position = 'bottom') +
  coord_flip()

distance_plot = bidders %>%
  pivot_longer(c(distrank_1, total_distrank), names_to = 'type', values_to = 'vals')%>%
  head(40) %>%
  ggplot(aes(x = bidder, y = vals, fill = type)) +
  geom_bar(position = 'stack', stat = 'identity') +
  labs(title = 'Top 20 Bidders Distance Rank is 1') +
  xlab('Bidder') +
  ylab('Bids') +
  scale_fill_discrete(name = "", labels = c("Total Bids", "Bids with Distance Rank = 1")) +
```

```
  theme(axis.title.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.text.y = element_blank(),
        legend.position = 'bottom') +
  coord_flip()

bidders_plot + distance_plot
```



We see that there is some difference between the rate at which the bidders are of distance rank 1 and the rate at which they win bids, but even so, the regression analysis shows no evidence of strategic bidding. What we can conclude from this graph is that bidding firms understand their cost structure and choose to bid only when they are very close to the site relative to others.

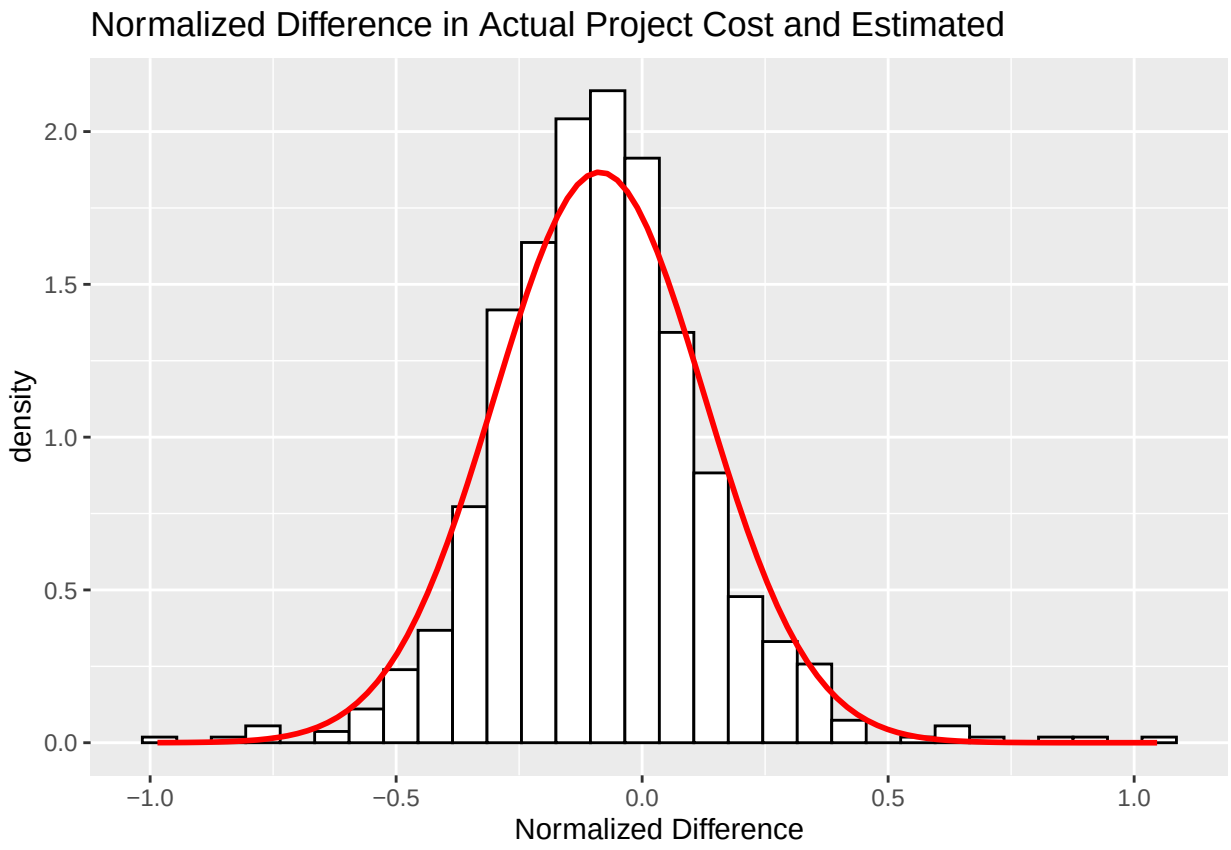
Extension 2

The second extension of the analysis is based on the fact that we see the fit of the data for the competitive model improves when we use the actual amount paid rather than the bid submitted. This leads us to wonder about the behavior of the engineering team in estimating project costs.

We examine whether the difference between the actual project cost and the engineering estimate normalized by the engineering estimate is normally distributed.

```
winners = df %>%
  filter(winner == 1) %>%
  mutate(bid_diff = (bidtotal_actual-engestimate)/engestimate)

bid_diffs = winners %$%
  bid_diff
```

```
mu = mean(bid_diffs)
sigma = sd(bid_diffs)

bid_diff_plot = winners %>%
  ggplot(aes(x=bid_diff)) +
  labs(title = 'Normalized Difference in Actual Project Cost and Estimated',
       x = 'Normalized Difference') +
  geom_histogram(colour="black",fill="white", aes(y = stat(density))) +
  stat_function(
    fun = dnorm,
    args = list(mean = mu, sd = sigma),
    lwd = 1,
    col = 'red'
  )

bid_diff_plot
```



We see that there is some potential fit between the normalized difference in the estimated project cost and the real project cost. We investigate this with a chi-squared test.

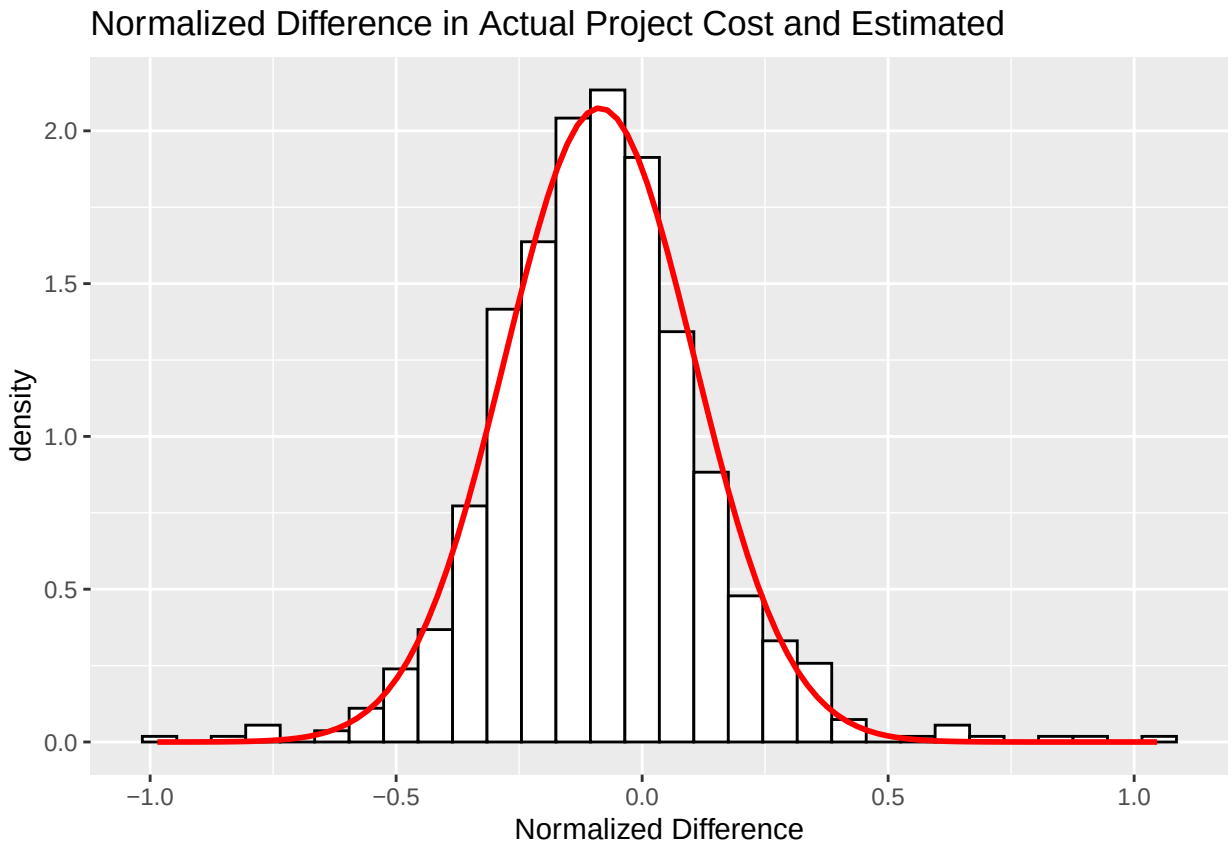
```
ChiSq <-function(Obs,Exp){
  sum((Obs-Exp)^2/Exp)
}

n_obs = nrow(winners)
dec = qnorm(seq(0.0, 1, by = 0.1), mu, sigma);   #11 bins
exp = rep(n_obs/10,10);       #expected persons per bin
bin_diffs <- numeric(10)
```

```
for (i in 1:10)
  bin_diffs[i] <- sum((bid_diffs >= dec[i]) & (bid_diffs <= dec[i+1]) )
obs = bin_diffs

test_stat = ChiSq(obs,exp)
p_val = pchisq(test_stat, df=7, lower.tail = FALSE)
```

We find a p_value of `0.0026` so, at a 1% level, we reject the null hypothesis that the normalized delta-estimate is normally distributed with mean `-0.0864` and standard deviation `0.2136`

However when we observe the graph we see that the estimated fit is really quite good but with less density around the mean than the data truly has. Let us then try to test whether a less spread normal distribution is in fact a good fit. We scale down the estimated variance by 90%.

```
sigma = sigma*0.9

bid_diff_plot_alt = winners %>%
  ggplot(aes(x=bid_diff)) +
  labs(title = 'Normalized Difference in Actual Project Cost and Estimated',
       x = 'Normalized Difference') +
  geom_histogram(colour="black",fill="white", aes(y = stat(density))) +
  stat_function(
    fun = dnorm,
    args = list(mean = mu, sd = sigma),
    lwd = 1,
    col = 'red'
  )

bid_diff_plot_alt
```




```
dec_alt = qnorm(seq(0.0, 1, by = 0.1), mu, sigma);   #11 bins
exp_alt = rep(n_obs/10,10);       #expected persons per bin
bin_diffs_alt <- numeric(10)
for (i in 1:10)
  bin_diffs_alt[i] <- sum((bid_diffs >= dec_alt[i]) & (bid_diffs <= dec_alt[i+1]) )
obs_alt = bin_diffs_alt

test_stat_alt = ChiSq(obs_alt,exp_alt)
p_val_alt = pchisq(test_stat_alt, df=7, lower.tail = FALSE)
```

We find a p_value of `0.2756` so we find, at a 1% level, we fail to reject the null hypothesis that the normalized delta-estimate is normally distributed with mean `-0.0864` and standard deviation `0.1923`.

We are now interested to get a 95% confidence interval for this mean. We know that the standard deviation is the estimated sigma scaled down by a factor of sqrt(n). We use the estimated mu as an estimate for the true mu.

```
upper_limit = mu + 1.96*sigma/sqrt(length(bid_diffs))
lower_limit = mu - 1.96*sigma/sqrt(length(bid_diffs))
```

We find that we are 95% confident that the true mean for our distribution is between `-0.0999` and `-0.0729`. This tells us that our true mean is very unlikely to be non-negative. In expectation, the engineering team at Caltrans provides an overestimate for project cost.