Microeconometrics: Problem Set 1

Due date: Next Class

Your answers should be produced in LaTeX, and should include all relevant graph and code. Code should be in the appropriate verbatim environment and properly documented. You are allowed to work in groups but you must turn in your own writeup. Submit your assignment via email to cconlon@stern.nyu.edu

Part 1: Guerre Perrigne and Vuong (Econometrica 2000)

This looks at a *first price sealed bid* (FPSB) auction. Bidders submit secret bids, and the highest bidder wins (and pays their bid). Given a valuation, bidders construct a bid $b_i(v_i)$ in order to maximize their expected profits. The bidder trades off: (a) the amount of surplus they receive conditional on winning $b_i(v_i) - v_i$; (b) the probability that they win $Pr(b_i(v_i) > b_j(v_j)|v_i)$ for all $j \neq i$.

Now let's calculate how to construct an optimal bid $b_i(v_i)$:

$$\pi(b) = (v_i - b_i)Pr(Win|b_i)$$
$$\pi(b) = (v_i - b_i)\prod_{i \neq j} Pr(b_i > b_j)$$

In other words, you only win if you submit the highest of N bids. There are a few tricks to simplify things even further.

1. Assume that the bidding function $b(v_i) = av_i$ is a scalar multiple of the bid.
2. Assume that values are drawn I.I.D from $v_i \sim U[0,1]$

The second equals sign uses the first assumption, the fourth equals sign uses the uniform distribution.

$$\pi(b) = (v_i - b_i)Pr(Win|b_i) = (v_i - b_i)Pr(b_i > av_j)^{N-1} = (v_i - b_i)Pr\left(v_i < \frac{b_i}{a}\right)^{N-1} = (v_i - b_i)\left(\frac{b_i}{a}\right)^{N-1}$$

Now – everything is easy. We just take $\max_{b_i} \pi(b_i)$ by taking the FOC:

$$0 = -(b_i/a)^{N-1} + (N-1)\frac{v_i - b_i}{a}(b_i/a)^{N-2}$$
$$b(v_i) = \frac{N-1}{N} v_i \qquad (1)$$

This tells us how to go from values to bids or vice versa. Notice that the *shading* is simple.

But.. there is no reason to believe that $v_i \sim U[0,1]$. We can use some other tricks (involving symmetry and the distribution of the highest bid among $N-1$ bidders $G(\max_j v_j)$ where $j \neq i$). Solving the FOC is more difficult (it involves a differential equation). The resulting solution has a nice form:

$$v_i = b_i + \frac{G(b_i)}{(N_i - 1)g(b_i)} \qquad (2)$$

Questions

I provided two datasets **auction_Xbidder.csv** that consist of either 3 bidder auctions or 4 bidder auctions.

1. Show that in (2) if $b_i = av_i$ and $v_i \sim U[0,1]$ that you get (1) back.
2. Plot a histogram for the bids in the three bidder auction and the four bidder auction separately (choose the number of bins with the "eyeball" test).
3. Calculate the mean and standard deviation for the three and four bidder auctions separately. Overlay the plot of the normal density on each histograms.
4. Construct a plot of the two normal densities and compare the mean and standard deviations, do they appear to be from the same distribution or not?
5. Calculate the `ecdf` of the bids for the three and four bidder auctions separately call this $\hat{G}(b)$.
6. Calculate the density of the bids for three and four bidder auctions separately and call this $\hat{g}(b)$.
7. Use $\hat{g}(b)$ and $\hat{G}(b)$ to construct an estimate for v_i which we call $\tilde{v}_i$.
   - Use both the Epanechnikov kernel $K(u) = \frac{3}{4}(1-u^2)\mathbf{1}(|u| < 1)$ and the uniform/rectangular kernel $K(u) = \mathbf{1}(|u| < 1)$ where $u = \frac{x_i - x}{h}$.
   - For each kernel vary the bandwidths $h = \{0.5, 0.1, 0.05, 0.01\}$
   - Plot each kernel and each bandwidth on top of the histogram of bids.
   - Comment on the performance of the procedure for different choices of kernel and bandwidth.
   - Try both the ECDF and the integrated density function for $\hat{G}(b)$. Which do you prefer and why?
8. Compute and plot separate ECDF for 3 and 4 bidder auctions. Plot these as *demand curves* (P on y-axis, Q on x-axis; hint: Q is the *fraction* of customers who would buy at price P).
9. If you can – report which choices a two-sample Kolmogorov Smirnoff test rejects that the two samples (three and four bidders) come from the same distribution. `R` will perform the test for you – but you may want to read up on how it works.

Part 2: Nonparametric Regression

Our goal here is to use **wholesaleShipments** to predict **retailSales** at a (non-random) sample of retail stores.

These data are fake but inspired by real data

1. Load the dataset **nonparametric_regression.csv** in `R`.
2. Plot the data with **wholsaleShipments** on the x axis and **retailSales** on the y-axis.
3. The data are highly skewed. Plot a histogram for each variable, and then repeat after taking the $\ln(x)$.
4. Re-plot the scatter plot on the log-scale.

```
df %>% ggplot2(aes(x=X, y=Y)) +
 scale_x_continuous(trans = 'log') +
 scale_y_continuous(trans = 'log')
```

5. Fit an ols regression (`lm`) on both the linear and the log scale.
   - Report regression results and R^2 in a table
   - Add the line of best fit to your (logged) scatterplot.
6. Fit the Nadaraya-Watson kernel regression on the same data.
7. Fit a Local-Linear regression on the same data.
   - Add both to your plots along with OLS
   - Hint: There are a number of `R` packages that will do nonparametric regression but `KernSmooth` is probably an easy place to start.
   - Hint: you may want to play with the bandwidth a bit here so that you don't get something crazy.

8. Now split your data into 10 folds.

```
require(caret)
# this gives the product ID's for each fold
flds <- createFolds(df$prod_id, k = 10, list = TRUE, returnTrain = FALSE)
```

9. For each fold do the following on the logged data only:

    - Estimate OLS
    $$\ln y \sim \beta_0 + \beta_1 \ln x + \varepsilon$$
    on the 90% of data **Not in your fold**.
    - Estimate Nadaraya-Watson
    $$\ln y \sim \beta(x) + \varepsilon$$
    on the 90% of data **Not in your fold**.
    - Estimate Local Linear
    $$\ln y \sim \beta_0(x) + \beta_1(x)\ln(x) + \varepsilon$$
    on the 90% of data **Not in your fold**.

10. Use your estimates to compute the mean-squared error **using the 10% of the data you withheld only**
$$\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

11. Repeat the exercise withholding each of the 10 folds (ten times).

    - How do the methods compare? Which has the lowest MSE?

12. Extra credit: use the 10 folds to select the bandwidth for the non-parametric regressions. Explain how you chose the bandwidth.