Data Science for Game Theory and Pricing

Instructor: Jacob LaRiviere, Director of Economics and Data Science at Amazon

Emails: lghhager@uw.edu (TA)

Course Assignments & Reading

Course assignments R Markdown files (.html) and turned in at the start of class on Canvas unless otherwise noted. Feel free to work in groups but everyone is required to turn in their own work with answers written in your own words. Keep in mind that in most cases a good answer is one precise sentence; quality is heavily favored over quantity. This will be graded on a full credit, half credit and no credit basis. All work must be typed.

At the beginning of each class the professors will lead a discussion around these questions. Students will be called on, potentially at random, to add their insight. This part of class will contribute heavily to your course participation grade.

Week 3, due Oct 16

Assignment to be turned in. Please turn in your Rmarkdown output and answers to the questions typed up and turned in on canvas. Make sure you do the theory portion of the assignment as well.

1) Let's focus on two variables HHLARGE ("fraction of households that are large") and EDUC ("fraction of shoppers with advanced education").
    a. What are the means and percentiles of each of these variables?
        HINT: `summary(oj$EDUC)`
    b. Using your coefficient estimates from the regression in Q9 of the previous problem set (if you did not include HHLARGE and EDUC, rerun the regression with them included):
        i. If we move from the median value of HHLARGE to the 75th percentile (3rd quartile), how much does log(quantity) change each week on average?
        HINT: using `coef(reg_output)["var_name"]` exports the coefficient on "var_name" from the regression model "reg_output". Alternatively, you can use the tidy() function from the broom package in R.
        Similarly, `summary(df$var_name)` will output a bunch of summary statistics for the variable var_name in data frame df. Using `summary(df$var_name)["3rd Qu."]` will take the level of the 3rd quantile from the summary of var_name.
        Note: if we wanted to assess the changes in levels, you'd want to take the exponent of everything.
        ii. If we move from the median value of EDUC to the 75th percentile (3rd quartile), how much does log(quantity) change each week on average?
        iii. Based on this analysis, which is the more important predictor of demand?
    c. Now let's see if these variables impact price sensitivity. Add two interaction terms (with logprice) to the model to test this.
        i. What are the coefficients on the interaction terms?
        ii. Does the sign of your estimates make sense based on your intuition?

iii. What are the coefficient estimates on the variables EDUC and HHLARGE that aren't part of the interaction term? How do they compare to your regression from 1b?
iv. Similar to 2b, if we move from the median value of each variable to the 3rd quartile, how much does elasticity change? Based on this, which is more important to price sensitivity?

d. You should notice that the coefficients on EDUC and HHLARGE have flipped sign once we include interaction terms with price. HHLARGE now appears to be a positive demand shifter and increases price sensitivity. Explain in words or pictures what is going on.

2) Create a new dataframe which takes the previous week's prices as a variable on the same line as the current week. This would enable you to see if there is *intertemporal* substitution.
   a. There are going to be a couple of steps. One way is using dplyr's lag() function (note that you'll want to group_by() the individual store and make sure the data is sorted by date). Alternatively, you can use the following procedure: create a new dataframe which is like the old one except that the week variable will change by a single week
      i. `Df1 <- oj`
      ii. `Df1$week <- Df1$week+1`
         1. This will replace week with week+1
      iii. The next step will use the `merge` function.
         1. `Df2 <- merge(oj, df1, by=c("brand","store","week"))`
         2. Investigate the Df2 and rename the lagged store values needed for a lagged price within the same store
   b. Now run a regression with this week's log(quantity) on current and last week's price.
   c. What do you notice about the previous week's elasticity? Does this make sales more or less attractive from a profit maximization perspective? Why?
3) In the last assignment you calculated the MSE on a test set. Let's expand that code to include 5-fold cross validation.
   a. Create 5 partitions of the data of equal size.
   b. Create 5 training datasets using 80% of the data for each one. This could be done by "appending" the data together using rbind or via sampling.
   c. Estimate a complex model using OLS which includes price, featured, brand, brand*price and lagged price, all the sociodemographic variables and interactions of EDUC and HHSIZE with price on each of the training sets and then calculate the MSE on the test sets using the predict command.
      i. Calculate the MSE for the model on the test set for each fold (e.g., there will be five sets of model parameters and five test set MSEs with 5-fold cross validation).
      ii. Average across the MSEs to get the cross validated MSE for an OLS model run on that particular set of features.
4) We haven't controlled for seasonality in our previous regressions. Let's change that.
   a. Use `ggplot` to plot logmove for each week (so week on the x-axis, logmove on the y-axis) as a scatterplot. Using the `alpha` parameter may help here to make the dots transparent and any trends more visible.
      i. Comment on your observations. Do you feel that logmove depends on week or not?
      ii. If week has an impact on sales, do you feel that effect is linear? A polynomial? Something else?

b. Let's use a nonparametric method to try to control for week. We'll do this by using the `loess` function from the `stats` package, which will fit a local polynomial. The key parameters for this function are the formula (the same as fitting a linear model using `lm`), the data, and the `span` parameter, which (for us) will be a value from 0-1. A value of .3 would indicate that we're using 30% of the closest points as our neighborhood.
c. Fit three loess models using span values of .05, .15, and .3 (the syntax should be `loess(logmove ~ week, data = oj, span = .05)`).
d. Plot the fitted values (see an example of this in the solutions for the last homework) on your plot in part (a) for each of the three loess models. Comment on how the choice of `span` impacts the model.
e. To use this in a regression, do the following:
    i. Take your predicted values from your favorite of the three models and compute the difference between logmove and the predicted value for each datapoint (that is, get the residual from the loess model). Another way to get these residuals if your model is called `my_loess_model` is to run `my_loess_model$residuals`.
    ii. Use these residuals as the LHS variable for a regression where we include brand and feat as interaction terms with log(price).
    iii. Compare the elasticity estimates you get from this regression to the same regression with logmove as the outcome (so where we're not accounting for any seasonality of sales). What are the similarities and differences? Which do you trust more, and why?

BONUS: Do (3) but using the same week's prices of *other* brand's OJ in the same regression. Here's a hint from base R: `dcast(oj_prices, store + week ~ brand)`. See if you can do the same but with `dplyr` or another R package. Try doing this for only a single brand of orange juice as a first step.