Untitled

Christopher Hayes, David Hakula, Filip Mellgren

2019-04-13

## R Markdown

# Step 0

df <- import("Cars.dta")  
df <- df %>% mutate(yearcountry = paste(year, country)) %>% as\_tibble()

# 1 Summary statistics

Show and discuss summary statistics of the car market in the five EU countries. Focus on what you believe are key variables given the questions at hand.

Could be the case: “Note that the panel is unbalanced because most models are not available throughout the entire period or in all countries.”

# Create two functions that are used throughout the analysis to create   
# descriptive tables  
  
# TODO: Add necessary summary stats. Do we need percentiles?  
summary\_stats <- function(x){  
 x <- as.matrix(x)  
   
 summary <- rbind(mean(x, na.rm = TRUE), sd(x, na.rm = TRUE), min(x, na.rm = TRUE),   
 max(x, na.rm = TRUE))  
 return(summary)  
}  
  
# Takes above and turns it into a nice table  
summary\_table<- function(summary){  
 summary <- as\_tibble(summary) %>% t() %>% round(2) %>% as\_tibble(rownames = "Variable") %>% rename ("Mean" = V1, "S.e." = V2, "Min." = V3, "Max." = V4)  
 return(summary)  
}

# Summary statistics to Q1.  
# TODO: Group by country? If so, stack them, keep the same columns  
descriptive\_table <- df %>%  
 select(country, year, domestic, qu, price, princ, horsepower, fuel, pop) %>%   
 mutate(pop = pop/1000000) %>%  
 map(function(x) summary\_stats(x)) %>% summary\_table() %>% mutate(country = "All")

## Warning: `as\_tibble.matrix()` requires a matrix with column names or a `.name\_repair` argument. Using compatibility `.name\_repair`.  
## This warning is displayed once per session.

for (c in 1:5) {  
 descriptive\_table <- rbind(descriptive\_table, df %>% filter(country == c) %>%   
 select(country, year, domestic, qu, price, princ, horsepower, fuel, pop) %>%   
 mutate(pop = pop/1000000) %>%  
 map(function(x) summary\_stats(x)) %>% summary\_table() %>% mutate(country = c))  
}  
  
  
  
tmp\_table <- descriptive\_table %>% arrange(Variable) %>%   
 mutate("se" = paste(S.e., ")", sep = ""),  
 "Mean\_se" = paste(Mean, se, sep = " (")) %>%   
 select(Variable, Mean\_se, country) %>% # Add min and max later for simplicity  
 spread(key = country, value = Mean\_se)  
  
tmp\_table <- rbind(tmp\_table, tmp\_table) %>% arrange(Variable) %>%   
 filter(Variable != "country") %>%   
 select(Variable, All, "1", "2", "3", "4", "5") %>% rename("Belgium" = "1",   
 "France" = "2",  
 "Germany" = "3",  
 "Italy" = "4",  
 "UK" = "5")  
  
tmp\_table\_means <- tmp\_table[seq(1, nrow(tmp\_table), 2), ] %>% mutate(All = word(All,1),  
 Belgium = word(Belgium),  
 France = word(France,1),  
 Germany = word(Germany,1),  
 Italy = word(Italy,1),  
 UK = word(UK,1))  
  
tmp\_table\_se <- tmp\_table[seq(1, nrow(tmp\_table), 2), ] %>% mutate(All = word(All,2),  
 Belgium = word(Belgium,2),  
 France = word(France,2),  
 Germany = word(Germany,2),  
 Italy = word(Italy,2),  
 UK = word(UK,2))  
minmax\_table <- df %>%  
 select(country, year, domestic, qu, price, princ, horsepower, fuel, pop) %>%   
 mutate(pop = pop/1000000) %>%  
 map(function(x) summary\_stats(x)) %>% summary\_table() %>%   
 select(Variable, Min., Max.) %>% filter(Variable != "country")  
  
   
final\_table <- rbind(tmp\_table\_means, tmp\_table\_se) %>% full\_join(minmax\_table) %>% arrange(Variable) %>%   
 select(Variable, All, Min., Max., Belgium, France, Germany, Italy, UK)

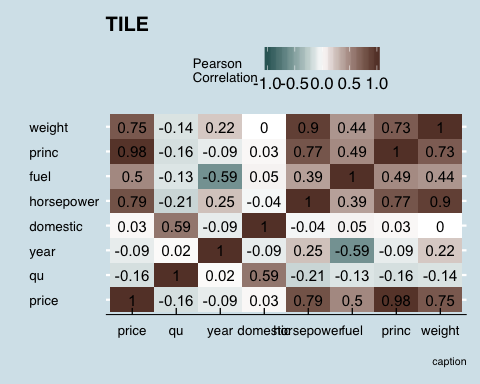
## Joining, by = "Variable"

xtable(final\_table)

## % latex table generated in R 3.5.2 by xtable 1.8-3 package  
## % Wed May 1 20:51:52 2019  
## \begin{table}[ht]  
## \centering  
## \begin{tabular}{rllrrlllll}  
## \hline  
## & Variable & All & Min. & Max. & Belgium & France & Germany & Italy & UK \\   
## \hline  
## 1 & domestic & 0.19 & 0.00 & 1.00 & 0 & 0.25 & 0.27 & 0.24 & 0.23 \\   
## 2 & domestic & (0.39) & 0.00 & 1.00 & (0) & (0.43) & (0.44) & (0.42) & (0.42) \\   
## 3 & fuel & 6.73 & 3.80 & 18.60 & 6.78 & 6.68 & 6.78 & 6.65 & 6.73 \\   
## 4 & fuel & (1.71) & 3.80 & 18.60 & (1.76) & (1.64) & (1.76) & (1.7) & (1.68) \\   
## 5 & horsepower & 57.26 & 13.00 & 169.50 & 56.78 & 56.29 & 57.46 & 57.3 & 58.54 \\   
## 6 & horsepower & (23.89) & 13.00 & 169.50 & (23.73) & (23.03) & (24.27) & (24.85) & (23.61) \\   
## 7 & pop & 48.06 & 9.66 & 82.06 & 9.92 & 55.36 & 68.47 & 56.59 & 57 \\   
## 8 & pop & (21.82) & 9.66 & 82.06 & (0.15) & (2.33) & (9.51) & (1.07) & (1.02) \\   
## 9 & price & 18.5 & 5.26 & 150.34 & 16.67 & 17.12 & 15.69 & 18.9 & 24.4 \\   
## 10 & price & (8.92) & 5.26 & 150.34 & (7.76) & (7.34) & (6.49) & (9.32) & (10.48) \\   
## 11 & princ & 0.83 & 0.24 & 6.47 & 0.73 & 0.75 & 0.65 & 0.99 & 1.05 \\   
## 12 & princ & (0.41) & 0.24 & 6.47 & (0.34) & (0.32) & (0.27) & (0.49) & (0.45) \\   
## 13 & qu & 19911.44 & 51.00 & 433694.00 & 3965.79 & 23430.26 & 31026.37 & 24345 & 19858.63 \\   
## 14 & qu & (37803.6) & 51.00 & 433694.00 & (4464.99) & (38164.69) & (50297.62) & (45817.56) & (29935.86) \\   
## 15 & year & 1985.43 & 1970.00 & 1999.00 & 1985.2 & 1985.31 & 1985.47 & 1985.74 & 1985.5 \\   
## 16 & year & (8.54) & 1970.00 & 1999.00 & (8.49) & (8.45) & (8.72) & (8.74) & (8.32) \\   
## \hline  
## \end{tabular}  
## \end{table}

Table 1 shows summary statistics for the five EU countries at a country level and at a European level. Perhaps the most interesting finding is that the average price of cars in the UK is well above the European average, at 24.4 compared to 18.5. We note that the average car sold in the UK is not drastically different from its European counterparts when it comes to physical attributes of the car, which makes the finding hard to rationalise. However, a potential explanation could be higher costs associated with the policy of driving on the right side of the road.

# Correlation matrix to Q1.  
# Strictly necessary?  
cormatdf <- df %>% select(price, qu, year, domestic, horsepower, fuel, princ, weight)  
cormat <- round(cor(cormatdf),2)  
  
melted\_cormat <- melt(cormat)  
  
melted\_cormat %>% ggplot(aes(x=Var1, y=Var2, fill=value)) +   
 geom\_tile() + theme\_economist() + scale\_colour\_economist() +  
 labs(title = "TILE", y = " ", x = " ",  
 caption = "caption") +  
 scale\_fill\_gradient2(low = "#336666", high = "#664033", mid = "white",   
 midpoint = 0, limit = c(-1,1), space = "Lab",   
 name="Pearson\nCorrelation") +  
 geom\_text(aes(Var1, Var2, label = value), color = "black", size = 4)

 Excluded co (model code), segment (ordinal variable), firm, country, because they are nonsensical Also height and width were excluded, because they were not deemed as important Pop was excluded because it is not a feature related to the cars

Discuss the difference in prices across countries. Look at UK mean prices for example.

Tendency to buy domestic cars

Mention the data set that we have access to (briefly)

# 2 Hedonic price regression

Table 2 shows a price regression with car characteristics as explanatory variables. Attributes that seem to be associated with a highe rcar price include horsepower and the width, reflecting that larger more powerful cars are more expensive.

We note that over the period, cars seem to have got less expensive, averaging at minus 1 percent per year according to the most elaborated specification with car model fixed effects.

# TODO: add the correct variables to X  
# TODO: Squeeze the latex table  
# TODO: Add segment dummies?  
  
X <- glue("year", "horsepower", "fuel", "width","height", "domestic", "year",   
 "country1", "country2", "country3", "country4", .sep = "+")  
  
formula <- glue("log(price) ~", X)  
lm.fit <- lm(formula, data = df)

fe\_formula <- glue("log(price) ~", X, "factor(co)", .sep = "+")  
fe.fit <- lm(fe\_formula, data = df)  
  
stargazer(lm.fit, fe.fit, type = "text", omit = "factor")

##   
## ==============================================================================  
## Dependent variable:   
## ----------------------------------------------------------  
## log(price)   
## (1) (2)   
## ------------------------------------------------------------------------------  
## year -0.020\*\*\* -0.011\*\*\*   
## (0.0003) (0.0003)   
##   
## horsepower 0.013\*\*\* 0.007\*\*\*   
## (0.0001) (0.0002)   
##   
## fuel -0.014\*\*\* -0.009\*\*\*   
## (0.001) (0.001)   
##   
## width 0.013\*\*\* 0.005\*\*\*   
## (0.0002) (0.0005)   
##   
## height -0.0002 0.002\*\*\*   
## (0.0003) (0.001)   
##   
## domestic 0.013\*\*\* -0.006\*\*   
## (0.004) (0.003)   
##   
## country1 -0.367\*\*\* -0.384\*\*\*   
## (0.004) (0.003)   
##   
## country2 -0.326\*\*\* -0.341\*\*\*   
## (0.004) (0.003)   
##   
## country3 -0.421\*\*\* -0.431\*\*\*   
## (0.004) (0.003)   
##   
## country4 -0.250\*\*\* -0.261\*\*\*   
## (0.004) (0.003)   
##   
## Constant 40.212\*\*\* 23.588\*\*\*   
## (0.520) (0.609)   
##   
## ------------------------------------------------------------------------------  
## Observations 11,483 11,483   
## R2 0.888 0.945   
## Adjusted R2 0.888 0.944   
## Residual Std. Error 0.145 (df = 11472) 0.103 (df = 11122)   
## F Statistic 9,087.120\*\*\* (df = 10; 11472) 534.118\*\*\* (df = 360; 11122)  
## ==============================================================================  
## Note: \*p<0.1; \*\*p<0.05; \*\*\*p<0.01

# 3 Market shares

Calculate the following market shares: (a) the market share of each car model, and (b) the market share of each car model in its segment. Discuss your findings. Assume that the total market size is the number of households, approximated by total population divided by 4

Comment: “In practice, the potential market size in a given year may be lower because cars are durable and consumers who just purchased a car may not consider buying a new one immediately.”

# TODO: maybe take the five highest and five lowest per a and b. Most popular cars and least popular car  
  
#a  
df <- df %>% mutate(msize = pop/4,   
 mshare = qu/msize,  
 ln\_mshare = log(mshare))  
# b  
df <- df %>% group\_by(segment, yearcountry) %>% mutate(tot\_s = sum(qu)) %>%  
 ungroup()  
df <- df %>% mutate(mshare\_s = qu / tot\_s,  
 ln\_mshare\_s = log(mshare\_s))  
  
# Table 1  
df %>% mutate(mshare = mshare \*100,  
 mshare\_s = mshare\_s \*100) %>%   
 select(mshare, mshare\_s, ln\_mshare, ln\_mshare\_s) %>%   
 map(function(x) summary\_stats(x)) %>% summary\_table()

## # A tibble: 4 x 5  
## Variable Mean S.e. Min. Max.  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 mshare 0.16 0.26 0 3.03  
## 2 mshare\_s 6.53 9.81 0.03 100   
## 3 ln\_mshare -7.38 1.5 -11.0 -3.5   
## 4 ln\_mshare\_s -3.64 1.47 -8.29 0

# market shares by firm over year, car models and countries  
# Likely not necessary  
df %>% group\_by(firm) %>% summarise(mean = mean(mshare)\*100, se = sd(mshare\*100),  
 min = min(mshare\*100), max = max(mshare\*100))

## # A tibble: 26 x 5  
## firm mean se min max  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 0.0716 0.0850 0.00238 0.365   
## 2 2 0.126 0.151 0.00200 0.736   
## 3 4 0.169 0.342 0.00184 3.03   
## 4 5 0.332 0.292 0.00249 1.39   
## 5 6 0.0643 0.0729 0.00199 0.425   
## 6 7 0.0270 0.0262 0.00207 0.159   
## 7 8 0.0830 0.114 0.00241 0.472   
## 8 9 0.0146 0.0117 0.00202 0.0522  
## 9 11 0.0651 0.0758 0.00205 0.409   
## 10 12 0.239 0.292 0.00223 1.36   
## # … with 16 more rows

To begin with, the market share of each car model is low, averaging at 0.16%, and no car model ever had a higher overal market share than the Fiat Uno, which in Italy 1986 held a 3.03% market share. Among the 24 observations where a market share above 2.5% were observed, 14 were Fiats sold in Italy. Another 7 were VW sold in Germany, and 2 were Renaults sold in France. These anecdotes reflect a broader pattern of car models being more popular in their country of origin, as noticed in the correlation between country and domestic (0.59).

In the respecitve segments, market shares tend to be much higher, averaging at 6.5% rather than 0.16%, and some markets experience extreme concentration within segments, as hinted by the maximum segment share observed being 100%.

We further investigated where extreme segment market shares are observed and found that the luxury segment is more often dominated by a few models. Notably, out of 91 models that experienced a market shares above 50%, 63 (69%) were in the luxury segment, and of the 626 models with at least 25% market share, 214 (34%) were in the luxury segment.

Overall, we expect segments to be important in the analysis as we expect consumers to be less likely to switch to car models between segments as a result of price changes on one model in a segment. This leaves room for market power as concentration within segments is typically higher, particularly in the luxury segment.

# 4 Logit demand

Supposed to pool the data?

Estimate demand using a logit specification (Berry, 1994). Use different estimators such as OLS and fixed effects. Use your estimated coefficients to calculate own-price elasticities. Interpret your results. Are there any interesting differences across firms?

df <- df %>% group\_by(yearcountry) %>% mutate(tot\_qu = sum(qu),  
 oshare = 1 - tot\_qu/msize) %>% ungroup()  
  
df <- df %>% mutate(ln\_dmshare = log(mshare) - log(oshare))

# Adds price to the X  
X <- glue("price", "year", "horsepower", "fuel", "width","height", "domestic", "year", .sep = "+")  
X\_fe <- glue(X, "country1", "country2", "country3", "country4", .sep = "+")  
  
formula <- glue("ln\_dmshare ~", X)  
fe\_formula <- glue("ln\_dmshare ~", X\_fe, "factor(co)", .sep = "+")  
lm.logit <- lm(formula, data = df)  
fe.logit <- lm(fe\_formula, data = df)  
  
# TODO: do we present this or is it enough with the elasticities?  
stargazer(lm.logit, fe.logit, type = "text", omit = "factor")

##   
## ============================================================================  
## Dependent variable:   
## --------------------------------------------------------  
## ln\_dmshare   
## (1) (2)   
## ----------------------------------------------------------------------------  
## price -0.021\*\*\* -0.051\*\*\*   
## (0.002) (0.003)   
##   
## year -0.002 -0.045\*\*\*   
## (0.002) (0.003)   
##   
## horsepower -0.033\*\*\* -0.011\*\*\*   
## (0.001) (0.001)   
##   
## fuel -0.057\*\*\* -0.056\*\*\*   
## (0.011) (0.010)   
##   
## width 0.061\*\*\* 0.062\*\*\*   
## (0.002) (0.004)   
##   
## height -0.001 0.030\*\*\*   
## (0.002) (0.005)   
##   
## domestic 1.632\*\*\* 1.852\*\*\*   
## (0.028) (0.023)   
##   
## country1 0.515\*\*\*   
## (0.034)   
##   
## country2 -0.447\*\*\*   
## (0.033)   
##   
## country3 -0.321\*\*\*   
## (0.036)   
##   
## country4 -0.269\*\*\*   
## (0.031)   
##   
## Constant -11.567\*\* 69.264\*\*\*   
## (4.492) (5.097)   
##   
## ----------------------------------------------------------------------------  
## Observations 11,483 11,483   
## R2 0.393 0.700   
## Adjusted R2 0.393 0.690   
## Residual Std. Error 1.170 (df = 11475) 0.836 (df = 11121)   
## F Statistic 1,062.749\*\*\* (df = 7; 11475) 71.748\*\*\* (df = 361; 11121)  
## ============================================================================  
## Note: \*p<0.1; \*\*p<0.05; \*\*\*p<0.01

# TODO: what coefficient do we use to estimate the elasticities? Maybe both  
alpha\_1 <- lm.logit[["coefficients"]][["price"]]  
df <- df %>% mutate(own\_price\_elasticity = alpha\_1 \* (1 - mshare)\*price)  
  
summary\_stats(df$own\_price\_elasticity) %>% summary\_table()

## # A tibble: 1 x 5  
## Variable Mean S.e. Min. Max.  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 V1 -0.39 0.19 -3.16 -0.11

# What's this good for?  
df <- df %>% group\_by(firm) %>% mutate(avg\_e\_jj\_l = mean(own\_price\_elasticity)) %>% ungroup()  
  
df <- fastDummies::dummy\_cols(df, select\_columns = "firm")

# 5 Nested logit demand

Estimate demand using a nested logit specification where segment is defined as the nest (Berry, 1994). Use different estimators such as OLS and fixed effects. Use your estimated coefficients to calculate own-price elasticities. Discuss your results.

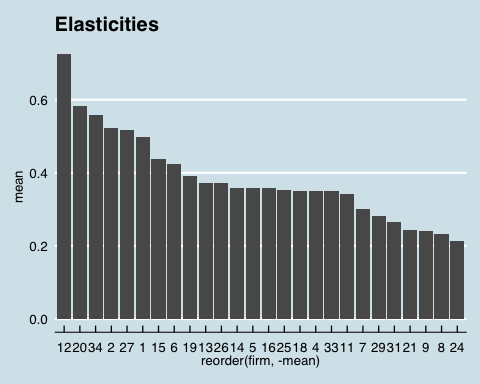
df <- df %>% mutate(ln\_mshare\_s = log(mshare\_s))  
  
  
# Adds ln\_mshare\_s to the X  
X <- glue("price", "year", "horsepower", "fuel", "width","height", "domestic",   
 "year", "ln\_mshare\_s", .sep = "+")  
X\_fe <- glue(X, "country1", "country2", "country3", "country4", .sep = "+")  
  
formula <- glue("ln\_dmshare ~", X)  
lm.nest <- lm(formula, data = df)  
fe\_formula <- glue("ln\_dmshare ~", X\_fe)  
fe.nest <- lm(fe\_formula, data = df)

# TODO: again, do what elasticfity do we calculate?  
alpha\_2 <- lm.nest[["coefficients"]][["price"]]  
sigma <- lm.nest[["coefficients"]][["ln\_mshare\_s"]]  
  
df <- df %>% mutate(e\_jj\_nl = alpha\_2 \* (1/(1-sigma) - (sigma/(1-sigma)) \* mshare\_s - mshare)\*price)  
  
# TODO: fix output  
df %>% select(e\_jj\_nl,own\_price\_elasticity) %>%  
 map(function(x) summary\_stats(x)) %>% summary\_table()

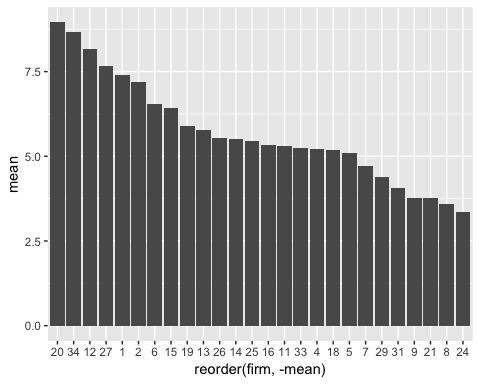
## # A tibble: 2 x 5  
## Variable Mean S.e. Min. Max.  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 e\_jj\_nl -5.72 2.76 -48.8 -1.57  
## 2 own\_price\_elasticity -0.39 0.19 -3.16 -0.11

# Calculates average average elasticity within nest for each firm  
df <- df %>% group\_by(firm) %>% mutate(avg\_e\_jj\_nl = mean(e\_jj\_nl)) %>% ungroup()

# TODO: Make it pretty!  
df %>% group\_by(firm) %>% summarise(mean = abs(mean(own\_price\_elasticity))) %>%   
 ggplot(aes(x = reorder(firm, -mean), y = mean)) +  
 geom\_bar(stat="identity") +  
 theme\_economist() + scale\_fill\_economist() + scale\_color\_economist() +  
 labs(title = "Elasticities")



df %>% group\_by(firm) %>% summarise(mean = abs(mean(e\_jj\_nl))) %>%   
 ggplot(aes(x = reorder(firm, -mean), y = mean)) +  
 geom\_bar(stat="identity")



# 6 Instruments

Goldberg, Verboven 2001 section 4.2 provides some insight: variables that shift the producers’ supply relations, but are excluded from the demand equations, make natural instruments

Suggest two types of instruments that could be used to solve the endogeneity problem of the price variable. Why do you propose these instruments?

Want to find something that shifts the supply curve: Things that change the marginal cost:

* price on car paint
* Strikes specific to the car industry or where strike wage remains.

To solve the endogeneity problem of the price variable, we need instruments that shifts the supply in market, without affecting the demand.

As a first instrument, we suggest price on car paint. This clearly affects the price through the supply side, by increasing the marginal costs of the supplier. The price on car paint hence should be relevant (note that this is testable, given data on price of car paint). It is unlikely that the price on car paint affects the market price through any channel on the demand (or any other channel for that matter) and hence it is likely that the instrument would also be valid.

Furthermore, we suggest to use data on strikes specific to the car industry. This should increase the production costs and hence shift supply, making the instrument relevant (not again that this is testable, given access to data). The validity of the instrument relies on the assumption that the strike does not affect the strikers demand for cars in such a way that total market demand is affected. We deem the validity as likely: the total population share working in the car industry is usually rather low on a national level. Also, strike funds are often utilized during strikes to compensate strikers for the absent wages.

# 7 Nash Bertrand

Assume Nash-Bertrand competition in prices on the supply side. Calculate the implied marginal costs, markups and Lerner index based on the estimates from the nested logit specification in [5]. Present firmlevel averages of price, marginal cost, markup and the Lerner index.

Do you think your results are economically meaningful? Explain and discuss.

# TODO: Note that markups are negative  
# markups  
df <- df %>% mutate(pderiv = alpha\_2 \* mshare\*((1/(1-sigma))-(sigma/(1-sigma))\*mshare\_s-mshare),  
 markup = -mshare/pderiv, # "-" to get the right sign  
 mc = price - markup,  
 lerner = markup/price)  
# TODO: what is markup2?   
  
# summarise

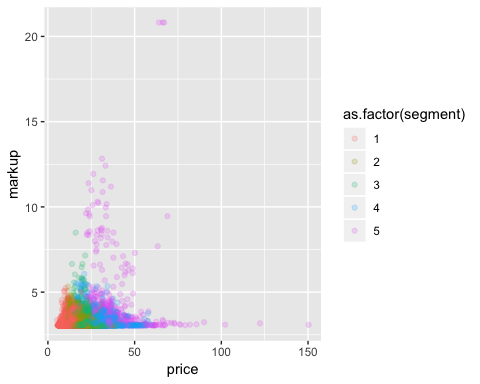
# firm level averages  
df <- df %>% group\_by(firm) %>% mutate(avg\_markup = mean(markup),  
 avg\_mc = mean(mc),  
 avg\_lerner = mean(lerner),  
 avg\_price = mean(price)) %>% ungroup()  
  
# Table for question 7. Pre merger descriptives.   
pricing\_table <- df %>% select(firm, avg\_price, avg\_mc, avg\_markup, avg\_lerner) %>%   
 distinct(firm, .keep\_all = TRUE) %>% arrange(-avg\_price)

Table X present firm level averages, ranked by average price. Apparently, the Lerner index is higher for low priced cars (the correlation between price and the lerner index is -82%). This finding can be justified when considering that prices observed are prices to consumers (in retail catalogues), not to car dealers which would be a better measure for the car manufacturers. We think the dealer would typically add something close to a constant fee per vehicle to cover her costs of selling it, absorbing most of the mark up observed for cheap cars, whereas larger cars with higher mark ups in absolute terms (correlation between markups and price is 17%) will still be able to maintain some markup after subtracting the fee. (Note to self: info about prices obtained form footnote 2 in Goldberg, Verboven 2001)

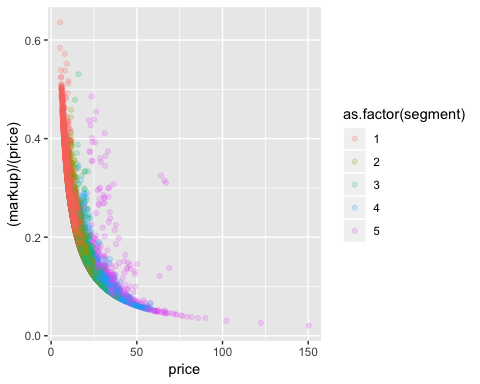
The results are obviously meaningful from an economic point of view insofar as the assumptions of the model hold. The average price is, as already discussed, the observed average price is what consumers paid for a car of a particular brand between 1970 and 1999 (in 1000s of euros in 1999 purchasing power), the marginal cost is the estimated marginal cost manufacturers face when producing the car model (in the same units) and the mark up is simply the difference between these. The average lerner index is the markups over price and a higher value indicate that a firm enjoys a higher market power, i.e. is either more efficient in production (cheaper marginal costs) or sells a product consumers are willing to pay a premium for.

The lerner index has another economic meaning, namely that the absolute value of the inverse lerner is the price elasticity of demand facing each firm. Derivation?

# For fun =)  
df %>% ggplot(aes(x = price, y = markup, color = as.factor(segment))) +  
 geom\_point(alpha = 0.2)



# subtracting a fixed fee such as the minimum markup (about 3), the relationship looks very different  
df %>% ggplot(aes(x = price, y = (markup)/(price), color = as.factor(segment))) +  
 geom\_point(alpha = 0.2)



# 8 Merger

post\_merger\_df <- function(dataframe, ratio\_mc){  
 # this takes a data frame, creates new variables and returns the same data frame  
 dataframe <- dataframe %>%   
 mutate(firmnew = if\_else(firm == 15 & country == 3 & year >= 1998, 26, firm),   
 segment\_new = if\_else(co == 177 & country == 3 & year >= 1998, 4, segment),   
 qu\_new = case\_when(   
 co == 164 & country == 3 & year >= 1998 ~ qu\*0.75,   
 co == 166 & country == 3 & year >= 1998 ~ qu\*0.5,   
 co == 168 & country == 3 & year >= 1998 ~ 0,   
 TRUE ~ qu),   
 mshare1 = qu\_new/msize)  
   
dataframe <- dataframe %>% group\_by(segment\_new, yearcountry) %>%   
 mutate(tot\_s1 = sum(qu\_new),   
 mshare\_s1 = qu\_new / tot\_s1) %>% ungroup()  
  
# TODO: double check we should be using alpha\_2 and not something else  
dataframe <- dataframe %>% mutate(mc = mc \* ratio\_mc,  
 pderiv\_new = alpha\_2 \* mshare1\*((1/(1-sigma))-(sigma/(1-sigma))\*mshare\_s1-mshare1),  
 markup\_new = -(mshare1 / pderiv\_new),  
 price\_new = mc + markup\_new,  
 lerner\_new = markup\_new / price\_new,  
 dprice = (price\_new-price)/price)  
  
# By firm: avg\_price, avg\_mc, avg\_markup, avg\_lerner  
dataframe <- dataframe %>% group\_by(firm) %>% mutate(avg\_markup\_new = mean(markup\_new),  
 avg\_mc\_new = mean(mc),  
 avg\_lerner\_new = mean(lerner\_new),  
 avg\_price\_new = mean(price\_new)) %>% ungroup()  
  
  
 return(dataframe)  
}

merger\_df <- post\_merger\_df(df, 1)

merger\_df %>% filter(year == 1998, country == 3) %>%  
 select(mshare\_s, mshare\_s1) %>% map(function(x) summary\_stats(x)) %>% summary\_table()

## # A tibble: 2 x 5  
## Variable Mean S.e. Min. Max.  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 mshare\_s 0.05 0.09 0 0.44  
## 2 mshare\_s1 0.05 0.09 0 0.48

## Post merger

# Compare per-merger and post-merger prices:   
# combine with table from 7  
pricing\_table\_new <- merger\_df %>% select(firm, avg\_price\_new) %>%   
 distinct(firm, .keep\_all = TRUE)  
  
full\_join(pricing\_table, pricing\_table\_new) %>%   
 mutate("Delta price (pct)" = 1000 \* (avg\_price\_new/avg\_price-1)) %>%  
 select(firm, avg\_price, avg\_price\_new, "Delta price (pct)", everything()) %>%   
 rename("Pre price" = avg\_price, "Post price" = avg\_price\_new) %>%   
 column\_to\_rownames(var = "firm") %>% xtable(digits = 2)

## Joining, by = "firm"

## % latex table generated in R 3.5.2 by xtable 1.8-3 package  
## % Wed May 1 20:52:06 2019  
## \begin{table}[ht]  
## \centering  
## \begin{tabular}{rrrrrrr}  
## \hline  
## & Pre price & Post price & Delta price (pct) & avg\\_mc & avg\\_markup & avg\\_lerner \\   
## \hline  
## 12 & 34.63 & 34.63 & 0.02 & 29.66 & 4.96 & 0.16 \\   
## 20 & 27.79 & 27.79 & 0.00 & 24.69 & 3.10 & 0.12 \\   
## 34 & 26.56 & 26.56 & 0.00 & 23.50 & 3.06 & 0.12 \\   
## 2 & 24.90 & 24.89 & -0.17 & 21.38 & 3.52 & 0.16 \\   
## 27 & 24.68 & 24.68 & -0.01 & 21.47 & 3.20 & 0.14 \\   
## 1 & 23.73 & 23.73 & 0.00 & 20.51 & 3.22 & 0.15 \\   
## 15 & 20.93 & & & 17.62 & 3.31 & 0.19 \\   
## 6 & 20.23 & 20.23 & -0.00 & 17.13 & 3.10 & 0.18 \\   
## 19 & 18.70 & 18.70 & 0.00 & 15.54 & 3.17 & 0.20 \\   
## 26 & 17.72 & 17.72 & 0.02 & 14.49 & 3.23 & 0.22 \\   
## 13 & 17.70 & 17.70 & 0.01 & 14.64 & 3.07 & 0.19 \\   
## 5 & 17.12 & 17.12 & 0.02 & 13.72 & 3.40 & 0.22 \\   
## 14 & 17.08 & 17.08 & 0.01 & 13.97 & 3.11 & 0.21 \\   
## 16 & 17.05 & 17.05 & 0.00 & 13.84 & 3.21 & 0.22 \\   
## 25 & 16.80 & 16.80 & 0.01 & 13.71 & 3.10 & 0.22 \\   
## 18 & 16.76 & 16.76 & 0.01 & 13.50 & 3.26 & 0.23 \\   
## 4 & 16.67 & 16.67 & -0.00 & 13.46 & 3.21 & 0.23 \\   
## 33 & 16.66 & 16.66 & 0.00 & 13.47 & 3.19 & 0.21 \\   
## 11 & 16.32 & 16.32 & 0.01 & 13.24 & 3.08 & 0.21 \\   
## 7 & 14.34 & 14.34 & 0.01 & 11.29 & 3.05 & 0.23 \\   
## 29 & 13.45 & 13.45 & 0.01 & 10.38 & 3.07 & 0.25 \\   
## 31 & 12.64 & 12.64 & 0.00 & 9.54 & 3.10 & 0.25 \\   
## 21 & 11.65 & 11.65 & 0.00 & 8.56 & 3.08 & 0.27 \\   
## 9 & 11.50 & 11.50 & 0.01 & 8.45 & 3.04 & 0.29 \\   
## 8 & 11.08 & 11.08 & 0.00 & 8.01 & 3.07 & 0.29 \\   
## 24 & 10.22 & 10.22 & 0.01 & 7.16 & 3.06 & 0.32 \\   
## \hline  
## \end{tabular}  
## \end{table}

ger\_table <- merger\_df %>% filter(year == 1998, country == 3) %>% drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x) summary\_stats(x)) %>% summary\_table() %>% cbind(name = "ger")  
  
# GM  
gm\_table <- merger\_df %>% filter(year == 1998, country == 3, firm == 15) %>% #drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x) summary\_stats(x)) %>% summary\_table() %>% cbind(name = "gm")  
  
# VW  
vw\_table <- merger\_df %>% filter(year == 1998, country == 3, firm == 26) %>% drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x) summary\_stats(x)) %>% summary\_table() %>% cbind(name ="vw")  
  
# Segments  
segment\_price\_change <- tibble()  
for (s in 1:5) {  
 segment\_price\_change <-   
 rbind(segment\_price\_change, merger\_df %>%   
 filter(year == 1998, segment == s) %>% drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x)   
 summary\_stats(x)) %>% summary\_table() %>% cbind(name =glue("segment",s)))  
}  
  
  
price\_change\_table <- rbind(ger\_table, gm\_table, vw\_table, segment\_price\_change) %>% select(Variable, Mean, name) %>% spread(key = Variable, value = Mean) %>% select(-dprice)   
  
price\_change\_table

## name price price\_new  
## 1 ger 15.35 15.35  
## 2 gm 19.92 19.04  
## 3 vw 17.18 17.20  
## 4 segment1 9.56 9.56  
## 5 segment2 13.73 13.73  
## 6 segment3 17.45 17.45  
## 7 segment4 24.74 24.72  
## 8 segment5 34.64 34.64

Note that dprice does not correspond to (price\_new - price) / price.

# 9 Efficient merger

Redo 8 with a changed mc. Tabulate all 5 segments.

merger\_df\_9 <- post\_merger\_df(df, 0.8)

merger\_df\_9 %>% filter(year == 1998, country == 3) %>%  
 select(mshare\_s, mshare\_s1) %>% map(function(x) summary\_stats(x)) %>% summary\_table()

## # A tibble: 2 x 5  
## Variable Mean S.e. Min. Max.  
## <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 mshare\_s 0.05 0.09 0 0.44  
## 2 mshare\_s1 0.05 0.09 0 0.48

ger\_table <- merger\_df\_9 %>% filter(year == 1998, country == 3) %>% drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x) summary\_stats(x)) %>% summary\_table() %>% cbind(name = "ger")  
  
# GM  
gm\_table <- merger\_df\_9 %>% filter(year == 1998, country == 3, firm == 15) %>% # Can't drop\_na()  
 select(price, price\_new, dprice) %>% map(function(x) summary\_stats(x)) %>% summary\_table() %>% cbind(name = "gm")  
  
# VW  
vw\_table <- merger\_df\_9 %>% filter(year == 1998, country == 3, firm == 26) %>% drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x) summary\_stats(x)) %>% summary\_table() %>% cbind(name ="vw")  
  
# Segments  
segment\_price\_change2 <- tibble()  
for (s in 1:5) {  
 segment\_price\_change2 <-   
 rbind(segment\_price\_change2, merger\_df\_9 %>%   
 filter(year == 1998, segment == s) %>% drop\_na() %>%   
 select(price, price\_new, dprice) %>% map(function(x)   
 summary\_stats(x)) %>% summary\_table() %>% cbind(name =glue("segment",s)))  
}  
  
  
  
  
price\_change\_table\_2 <- rbind(ger\_table, gm\_table, vw\_table, segment\_price\_change2) %>% select(Variable, Mean, name) %>%   
 spread(key = Variable, value = Mean) %>% select(-dprice)  
  
price\_change\_table\_2 <- price\_change\_table\_2 %>% rename(efficient\_price = price\_new)

# Merge the two price change tables  
pre\_post\_table <- full\_join(price\_change\_table, price\_change\_table\_2)

## Joining, by = c("name", "price")

pre\_post\_table <- pre\_post\_table %>% gather(price\_type, price, -name) %>%   
 mutate(treatment = if\_else(price\_type == "price", 0, 1)) # WHy do we do this?

# 10 2 more features

We know what happens to production, but happens to sales? What branding effects are there that might affect sales?

More nests?

The analysis conducted above focuses on the company level and production aspect of the merger. However, other important aspects of the businesses involved are likely to be affected as well.

One aspect is sales and marketing. The car market is likely to be characterized by consumer loyalties, that perhaps is not captured in the product characteristics analysis made, above that could be affected by a merger: It is not curtain (as the above conducted analysis assumes) that the demand for a particular brand stays the same. E.g., some customers might value the “Americaness” and the American values that GM symbolizes. Such customers might be less willing to buy their models after a merger, where the GM production has been taken over by German Volkswagen. To the contrary, it could also create interesting co-branding opportunities. Such aspects of sales and marketing should hence also be considered in merger analysis. This could for example be done by incorporating these aspects into the model above, by scoring of these aspects of the products and estimation of how they would change with the merger through consumer surveys.

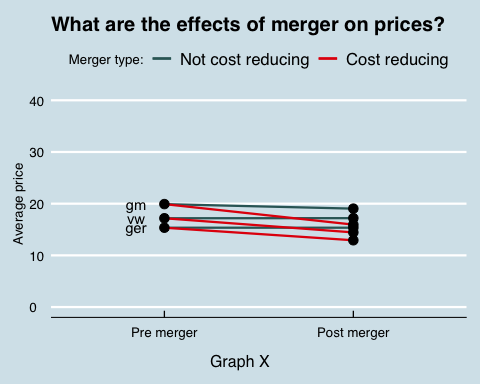
Another aspect to be considered is the number of nests. We believe that a model using a richer variety of nests would more accurately capture the market behavior. [Filip: I don’t remember what we said here?]

Taxes varying across nation? or is it captured by country dummies?

pre\_post\_table <- pre\_post\_table %>% select(-treatment) %>%   
 spread(price\_type, price)

graph\_table <- pre\_post\_table %>% filter(name %in% c("ger", "gm", "vw"))  
  
graph\_table %>%  
 ggplot(aes(x = as.factor("Pre merger"), y = price)) +  
 geom\_segment(aes(xend = as.factor("Post merger"), yend = price\_new,color = "#336666"),  
 size = 0.8) +  
 geom\_segment(aes(xend = as.factor("Post merger"), yend = efficient\_price, color = "#e3120b"), size = 0.8) +  
 geom\_point(size = 3) +  
 geom\_point(aes(x = as.factor("Post merger"),y = price\_new), size = 3) +  
 geom\_point(aes(x = as.factor("Post merger"), y = efficient\_price), size = 3) +  
 theme\_economist() + scale\_fill\_economist() + scale\_color\_economist() +  
 annotate("text", x = 0.85, y = graph\_table$price, label = as.character(graph\_table$name)) +  
 labs(title = "What are the effects of merger on prices?", y = "Average price",  
 tag = "Graph X") +  
 scale\_x\_discrete(name ="", limits=c("Pre merger","Post merger")) +  
 scale\_colour\_manual(guide = "legend", name = "Merger type:",  
 values =c("#336666"="#336666","#e3120b"="#e3120b"),   
 labels = c("Not cost reducing","Cost reducing")) +  
 theme(plot.tag.position = "bottom") +  
 scale\_y\_continuous(limits=c(0, 40))

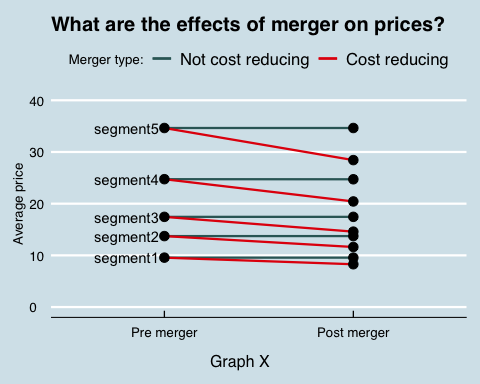
## Scale for 'colour' is already present. Adding another scale for  
## 'colour', which will replace the existing scale.



ggsave("merger\_causal\_effect.png", height = 20, width = 12, units = "cm")

# Graph above, but for segments.   
  
graph\_table <- pre\_post\_table %>% filter(!name %in% c("ger", "gm", "vw"))  
  
graph\_table %>%  
 ggplot(aes(x = as.factor("Pre merger"), y = price)) +  
 geom\_segment(aes(xend = as.factor("Post merger"), yend = price\_new,color = "#336666"),  
 size = 0.8) +  
 geom\_segment(aes(xend = as.factor("Post merger"), yend = efficient\_price, color = "#e3120b"), size = 0.8) +  
 geom\_point(size = 3) +  
 geom\_point(aes(x = as.factor("Post merger"),y = price\_new), size = 3) +  
 geom\_point(aes(x = as.factor("Post merger"), y = efficient\_price), size = 3) +  
 theme\_economist() + scale\_fill\_economist() + scale\_color\_economist() +  
 annotate("text", x = 0.8, y = graph\_table$price, label = as.character(graph\_table$name)) +  
 labs(title = "What are the effects of merger on prices?", y = "Average price",  
 tag = "Graph X") +  
 scale\_x\_discrete(name ="", limits=c("Pre merger","Post merger")) +  
 scale\_colour\_manual(guide = "legend", name = "Merger type:",  
 values =c("#336666"="#336666","#e3120b"="#e3120b"),   
 labels = c("Not cost reducing","Cost reducing")) +  
 theme(plot.tag.position = "bottom") +  
 scale\_y\_continuous(limits=c(0, 40))

## Scale for 'colour' is already present. Adding another scale for  
## 'colour', which will replace the existing scale.



ggsave("merger\_causal\_effect\_segment.png", height = 20, width = 12, units = "cm")