Economía Computacional: Tarea 1

Isidoro Garcia

2021

```
library(tidyverse)
library(data.table)
library(RCT)
library(knitr)
library(lfe)
library(broom)
```

En esta tarea pondrán en práctica los conceptos de High Dimensional Inference y Regresión. La base de datos muestra las compras de helados Ben & Jerry. Cada fila es una compra. Cada columna es una característica del helado comprado o de la persona que compró.

Limpieza de datos

Carga los datos en BenAndJerry.csv.

```
# Carga la base de datos
base<-fread(list.files(pattern = '.csv'))
```

1. Cuales son las columnas de la base? Muestra una tabla con ellas

```
kable(names(base))
```

x
quantity
price_paid_deal
price_paid_non_deal
coupon_value
promotion_type
size1_descr
flavor_descr
formula_descr
household_id
household_size
household_income
age_of_female_head
age_of_male_head
age_and_presence_of_children
male_head_employment
female_head_employment
male_head_education
female_head_education
marital_status
male_head_occupation

x
female_head_occupation
household_composition
race
hispanic_origin
region
scantrack_market_identifier
fips_state_code
fips_county_code
type_of_residence
kitchen_appliances
tv_items
female_head_birth
male_head_birth
household_internet_connection

2. A qué nivel está la base? Esto es, cuál es la variable que define la base de manera única. Si no la hay, crea una y muestra que es única a nivel de la base (Muestra el código)

```
nrow(base) # Obs en la base
```

```
## [1] 21974
```

```
# unicas por variable
(unicas<-map_dbl(base %>% select_all(), ~n_distinct(.)))
```

```
##                 quantity              price_paid_deal
##                       13                          562
##       price_paid_non_deal                 coupon_value
##                      575                          198
##           promotion_type                  size1_descr
##                        5                            2
##             flavor_descr                formula_descr
##                       50                            2
##             household_id               household_size
##                     6385                            9
##         household_income            age_of_female_head
##                       19                           10
##          age_of_male_head  age_and_presence_of_children
##                       10                            8
##      male_head_employment       female_head_employment
##                        5                            5
##       male_head_education        female_head_education
##                        7                            7
##           marital_status         male_head_occupation
##                        4                           12
##    female_head_occupation        household_composition
##                       13                            7
##                     race              hispanic_origin
##                        4                            2
##                   region  scantrack_market_identifier
##                        4                           53
##          fips_state_code             fips_county_code
##                       49                          178
```

```
##             type_of_residence           kitchen_appliances
##                             7                            9
##                      tv_items            female_head_birth
##                             4                          140
##               male_head_birth household_internet_connection
##                           133                            2
```

```r
# Creando el primary key
base <-
  base %>%
  mutate(primary_key = row_number())
```

3. Que variables tienen valores vacíos? Haz una tabla con el porcentaje de vacíos para las columnas que tengan al menos una observación vacía

```r
var_na<-map_dbl(base %>% select_all(),
              ~100*sum(is.na(.))/nrow(base))

var_na<-var_na[var_na>0]

kable(var_na)
```

	x
promotion_type	59.0698098
female_head_occupation	10.3167380
scantrack_market_identifier	18.5127878
tv_items	0.1547283

4. Haz algo con los valores vacíos (Se deben reemplazar por algún valor? Eliminar de la base?). Justifica tu respuesta.

- Promotion type po no promotion porque parece obvio que el vacío significa no promoción.
- En female occupation y market identifier no es una respuesta obvia. Dado que ademas su nivel de NA's son muchos para filtrar, se pueden hacer dos cosas: 1) Declarar explícitamente los NA's como 'Other' o 2) Quitar las columnas.
- Finalmente, para el codigo del county y para el número de televisiones, las respuestas tampoco son obvias. Dado que son menos del 1 por ciento de la base, las filtro.

```r
# promotion_type
table(base$promotion_type, useNA = 'ifany')
```

```
##
##     1     2     3     4  <NA>
##  6509  1106  1258   121 12980
```

```r
# Reemplazando por 'no promoción'
base<-
  base %>%
  mutate(promotion_type = replace_na(promotion_type, replace = 'no promotion'))

table(base$promotion_type, useNA = 'ifany')
```

```
##
##            1            2            3            4 no promotion
```

```
##          6509          1106          1258           121         12980
```

```r
# female_head_education
table(base$female_head_occupation, useNA = 'ifany')
```

```
## 
##    1    2    3    4    5    6    7    8    9   10   11   12 <NA> 
## 5225 2685 2146 1188  251  363   43 1133   22  166   33 6452 2267
```

```r
# scan market identifier
table(base$scantrack_market_identifier, useNA = 'ifany')
```

```
## 
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16 
##  960  609  269  196  122  118  988  559  310  229  259  802  650  468  136  345 
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32 
##  442  666  567  424  137  394  187  569  318  332  199  382  350  240  105  337 
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48 
##  406  128  102  138  137  472  311  200  392  499  208  404   79  259  117   72 
##   49   50   51   52 <NA> 
##  251  468  403  191 4068
```

```r
# Reemplazando por 'other' en la female_head_occupation y market identifier
base <-
  base %>%
  mutate(female_head_occupation = replace_na(female_head_occupation, replace = 'Other'),
         scantrack_market_identifier = replace_na(scantrack_market_identifier, replace = 'Other'))

table(base$female_head_occupation, useNA = 'ifany')
```

```
## 
##     1    10    11    12     2     3     4     5     6     7     8     9 Other 
##  5225   166    33  6452  2685  2146  1188   251   363    43  1133    22  2267
```

```r
table(base$scantrack_market_identifier, useNA = 'ifany')
```

```
## 
##     1    10    11    12    13    14    15    16    17    18    19     2    20 
##   960   229   259   802   650   468   136   345   442   666   567   609   424 
##    21    22    23    24    25    26    27    28    29     3    30    31    32 
##   137   394   187   569   318   332   199   382   350   269   240   105   337 
##    33    34    35    36    37    38    39     4    40    41    42    43    44 
##   406   128   102   138   137   472   311   196   200   392   499   208   404 
##    45    46    47    48    49     5    50    51    52     6     7     8     9 
##    79   259   117    72   251   122   468   403   191   118   988   559   310 
## Other 
##  4068
```

```r
# Census county code y tv_items: eliminar esas filas
table(base$tv_items, useNA = 'ifany')
```

```
## 
##    1    2    3 <NA> 
## 7986 7530 6424   34
```

```r
base<-
  base %>%
  filter(!is.na(tv_items))
```

```r
var_na<-map_dbl(base %>% select_all(),
              ~100*sum(is.na(.))/nrow(base))

var_na<-var_na[var_na>0]
```

5. Muestra una tabla de estadisticas descriptivas de la base. Esta debe tener cada columna númerica con algunas estadísticas descriptivas (N, media, min, p05, p25, p50, p75, p90, p95, max).

```r
est_desc<-summary_statistics(base)
kable(est_desc, digits = 2)
```

variable	mean	n	0	0.05	0.1	0.25	0.5	0.75	0.9	0.95	1
quantity	1.28	21940	1	1.00	1.0	1.00	1.00	1.00	2.00	2.00	21.00
price_paid_deal	1.74	21940	0	0.00	0.0	0.00	0.00	3.34	4.50	6.86	28.88
price_paid_non_deal	2.45	21940	0	0.00	0.0	0.00	2.99	3.55	4.99	6.86	69.72
coupon_value	0.16	21940	0	0.00	0.0	0.00	0.00	0.00	0.50	1.00	12.95
household_id	16618328.05	21940	2000358	2054629.00	2099686.8	2143518.75	2501765.00	3018433.00	3033901.00	3038840.00	3040689.00
household_size	2.46	21940	1	1.00	1.0	2.00	2.00	3.00	4.00	5.00	9.00
household_income	21.48	21940	3	11.00	13.0	17.00	23.00	26.00	27.00	28.00	30.00
age_of_female_head	5.51	21940	0	0.00	0.0	4.00	6.00	8.00	8.00	9.00	9.00
age_of_male_head	4.76	21940	0	0.00	0.0	2.00	5.00	8.00	8.00	9.00	9.00
age_and_presence_of_children	7.40	21940	1	2.00	2.0	6.00	9.00	9.00	9.00	9.00	9.00
male_head_employment	3.09	21940	0	0.00	0.0	1.00	3.00	3.00	9.00	9.00	9.00
female_head_employment	4.20	21940	0	0.00	0.0	2.00	3.00	9.00	9.00	9.00	9.00
male_head_education	3.32	21940	0	0.00	0.0	2.00	4.00	5.00	6.00	6.00	6.00
female_head_education	3.98	21940	0	0.00	0.0	3.00	4.00	5.00	6.00	6.00	6.00
marital_status	1.94	21940	1	1.00	1.0	1.00	1.00	3.00	4.00	4.00	4.00
male_head_occupation	5.11	21940	1	1.00	1.0	1.00	4.00	8.00	12.00	12.00	12.00
household_composition	2.57	21940	1	1.00	1.0	1.00	1.00	5.00	7.00	7.00	8.00
race	1.24	21940	1	1.00	1.0	1.00	1.00	1.00	2.00	3.00	4.00
hispanic_origin	1.95	21940	1	2.00	2.0	2.00	2.00	2.00	2.00	2.00	2.00
region	2.63	21940	1	1.00	1.0	2.00	3.00	4.00	4.00	4.00	4.00
fips_state_code	27.19	21940	1	6.00	6.0	12.00	26.00	39.00	48.00	53.00	56.00
fips_county_code	79.73	21940	1	3.00	7.0	25.00	59.00	101.00	163.00	201.00	810.00
type_of_residence	2.08	21940	1	1.00	1.0	1.00	1.00	2.00	5.00	6.00	7.00
kitchen_appliances	3.81	21940	1	1.00	1.0	4.00	4.00	4.00	7.00	7.00	9.00
tv_items	1.93	21940	1	1.00	1.0	1.00	2.00	3.00	3.00	3.00	3.00
household_internet_connection	1.16	21940	1	1.00	1.0	1.00	1.00	1.00	2.00	2.00	2.00
primary_key	10986.39	21940	1	1098.95	2195.9	5490.75	10983.50	16479.25	19780.10	20877.05	21974.00

6. Hay alguna númerica que en verdad represente una categorica? Cuales? Cambialas a factor

Las variables númericas que en verdad son factores son:

- marital_status
- male_head_occupation
- age_and_presence_of_children
- female/male_head_employment

- male/female_head_education
- household_composition
- race
- hispanic
- region
- fips_state_code
- fips_county_code
- type_of_residence
- household_internet_connection

```
base<-
  base %>%
  mutate(marital_status = factor(marital_status, levels = 1:4),
         male_head_occupation = factor(male_head_occupation, levels = 1:12),
         age_and_presence_of_children = factor(age_and_presence_of_children, levels = 1:9),
         male_head_employment = factor(male_head_employment, levels = 0:9),
         female_head_employment = factor(female_head_employment, levels = 0:9),
         household_composition = factor(household_composition, levels = 1:8),
         race = factor(race, levels = 1:4),
         hispanic_origin = hispanic_origin -1,
         region = factor(region, levels = 1:4),
         fips_state_code = factor(fips_state_code, levels = 1:56),
         fips_county_code = factor(fips_county_code, levels = 1:810),
         type_of_residence = factor(type_of_residence, levels = 1:7),
         household_internet_connection = household_internet_connection -1)
```

7. Revisa la distribución de algunas variables. Todas tienen sentido? Por ejemplo, las edades?

```
base<-
  base %>%
  mutate(age_of_female_head = if_else(age_of_female_head<16, 16, as.double(age_of_female_head)),
         age_of_male_head = if_else(age_of_male_head<16, 16, as.double(age_of_male_head)))
```

8. Finalmente, crea una variable que sea el precio total pagado y el precio unitario

```
base<-
  base %>%
  mutate(price = price_paid_deal + price_paid_non_deal,
         price_unit = price / quantity)
```

Exploración de los datos

Intentaremos comprender la elasticidad precio de los helados. Para ello, debemos entender:

- La forma funcional base de la demanda (i.e. como se parecen relacionarse q y p).
- Qué variables irían en el modelo de demanda y cuáles no para encontrar la elasticidad de manera 'insesgada'.
- Qué variables cambian la relacion de q y p. Esto es, que variables alteran la elasticidad.

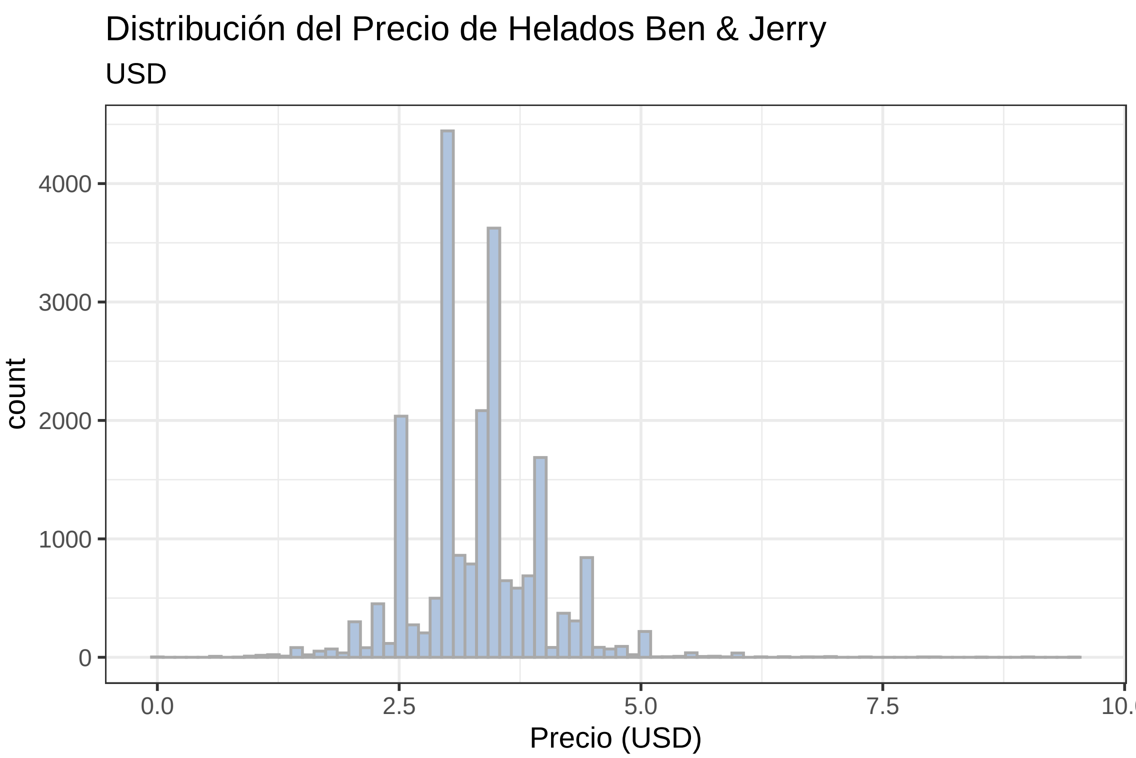
Algo importante es que siempre debemos mirar primero las variables más relevantes de cerca y su relación en:

- Relación univariada
- Relaciones bivariadas
- Relaciones trivariadas

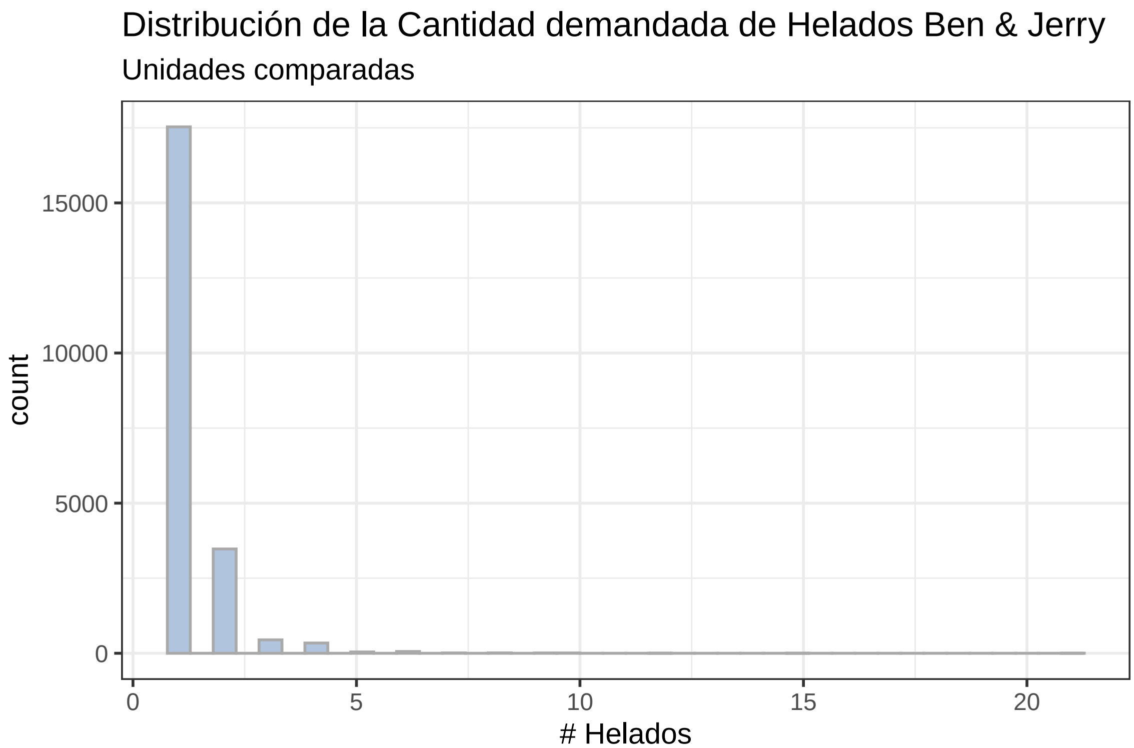
Importante: Las gráficas deben estar bien documentadas (título, ejes con etiquetas apropiadas, etc). Cualquier gráfica que no cumpla con estos requisitos les quitaré algunos puntos.

9. Cómo se ve la distribución del precio unitario y de la cantidad demandada. Haz un histograma.

```
ggplot(base, aes(price_unit))+
  geom_histogram(fill = 'lightsteelblue', color = 'darkgrey', bins = 80)+
  theme_bw()+
  labs(title = 'Distribución del Precio de Helados Ben & Jerry',
       subtitle = 'USD',
       x = 'Precio (USD)')
```



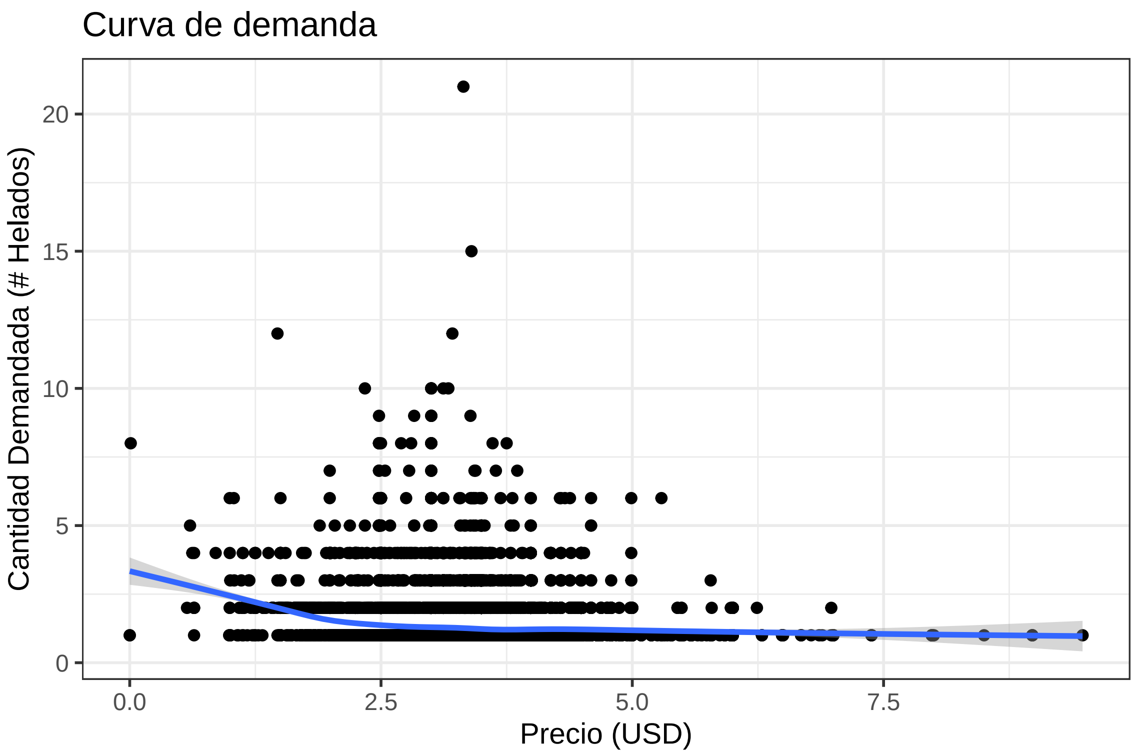

```
ggplot(base, aes(quantity))+
  geom_histogram(fill = 'lightsteelblue', color = 'darkgrey', bins = 40)+
  theme_bw()+
  labs(title = 'Distribución de la Cantidad demandada de Helados Ben & Jerry',
       subtitle = 'Unidades comparadas',
       x = '# Helados')
```



10.

Grafica la $q(p)$. Que tipo de relación parecen tener?

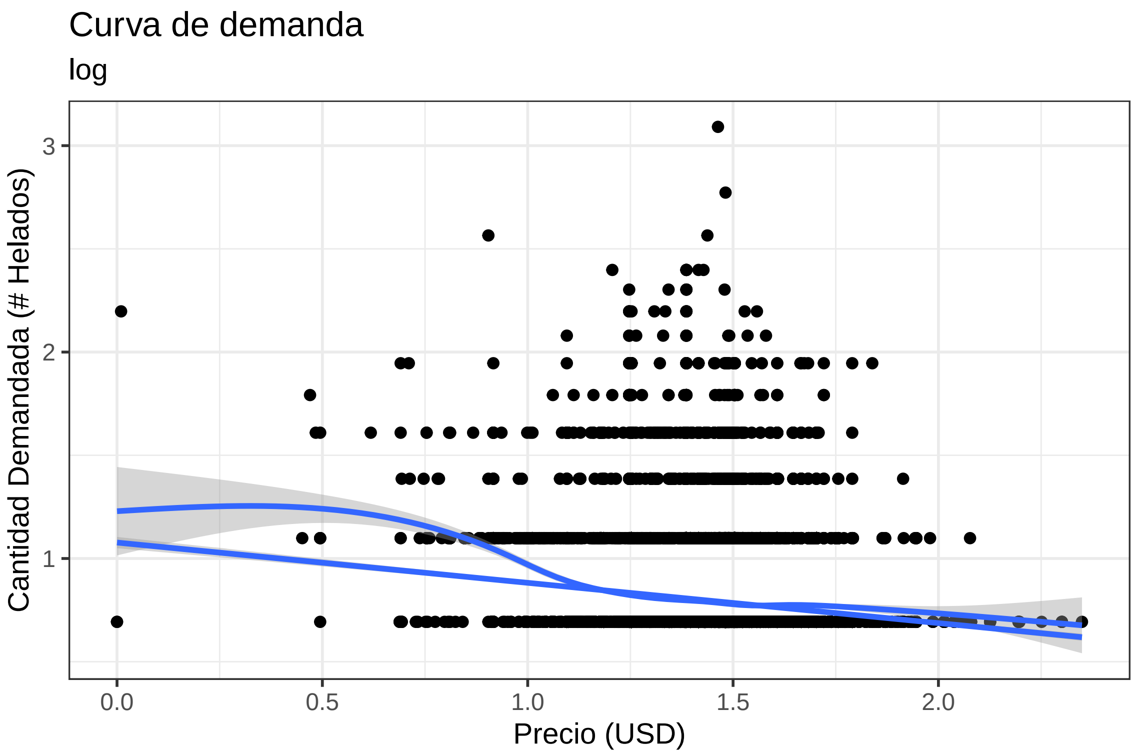
```
ggplot(base, aes(price_unit, quantity))+
  geom_point()+
  geom_smooth()+
  theme_bw()+
  labs(title = 'Curva de demanda',
       y = 'Cantidad Demandada (# Helados)',
       x = 'Precio (USD)')
```



11.

Grafica la misma relación pero ahora entre $log(p+1)$ y $log(q+1)$

```
ggplot(base, aes(log(price_unit+1), log(quantity+1)))+
  geom_point()+
  geom_smooth()+
  geom_smooth(method = 'lm')+
  theme_bw()+
  labs(title = 'Curva de demanda',
       subtitle = 'log',
       y = 'Cantidad Demandada (# Helados)',
       x = 'Precio (USD)')
```

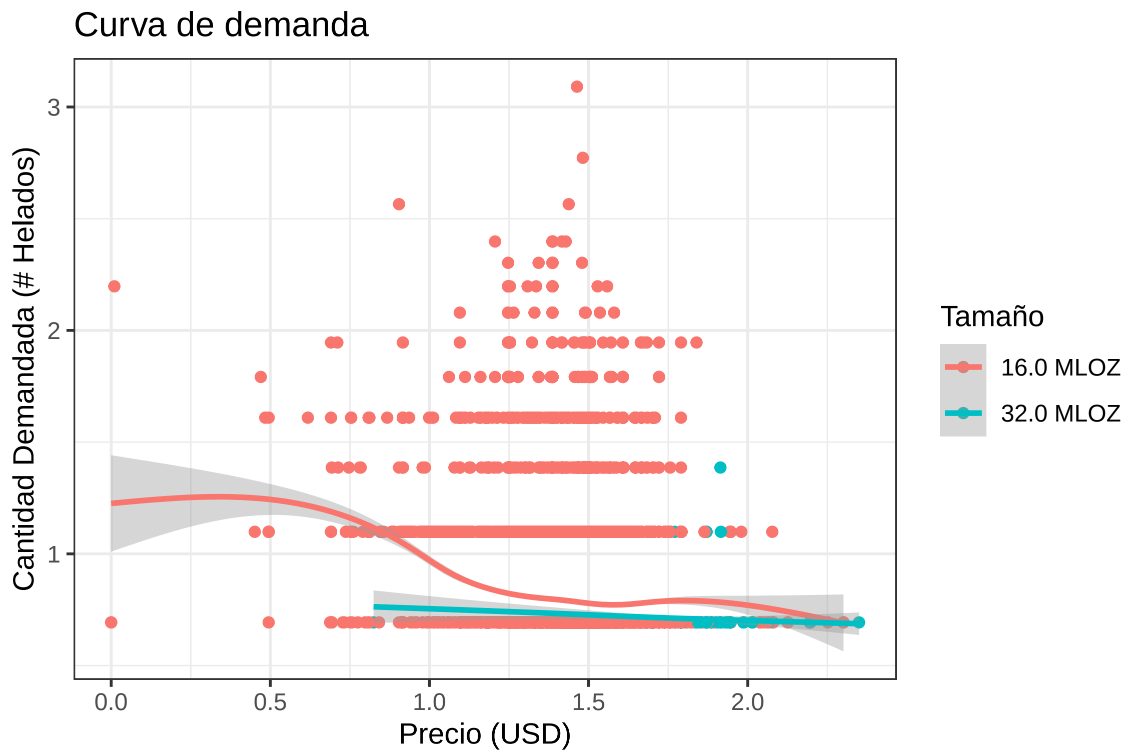


Usemos la transformación logarítmica a partir de este punto. Grafiquemos la demanda inversa.

12. Grafica la curva de demanda por tamaño del helado. Parece haber diferencias en la elasticidad precio dependiendo de la presentación del helado? (2 pts)

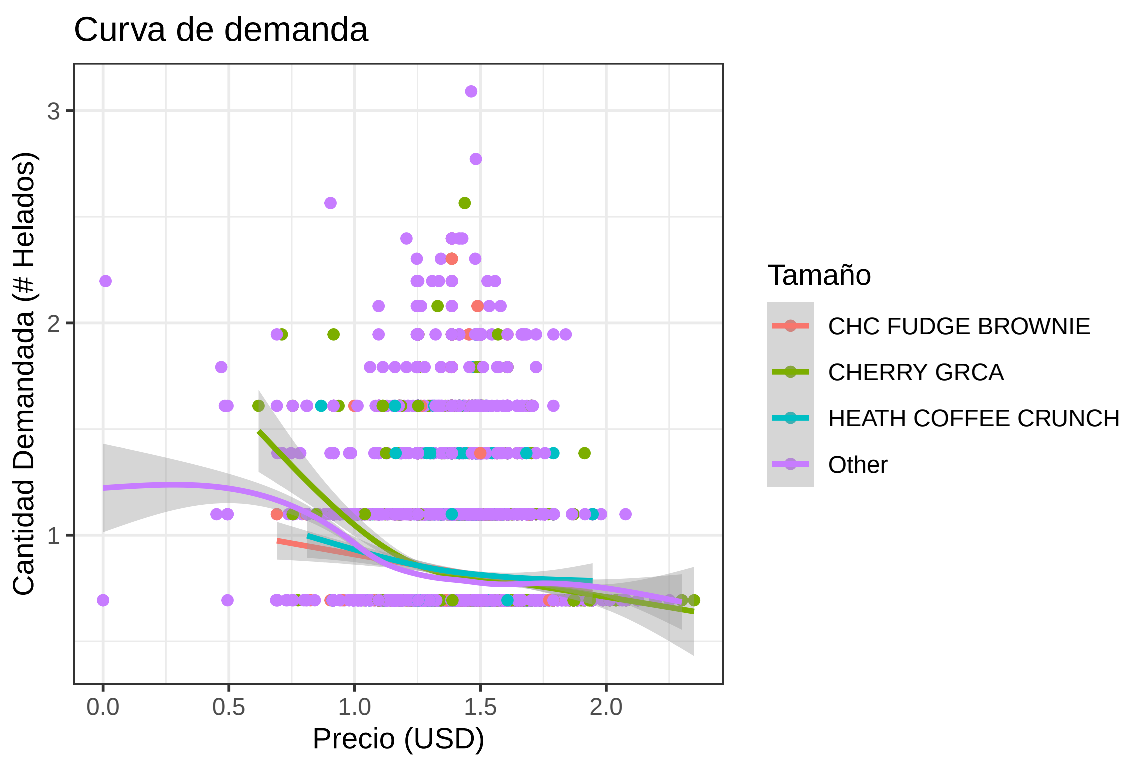
La demanda por helados de mayor tamaño (32 OZ) parecen tener una demanda mas inelástica.

```
ggplot(base, aes(log(price_unit+1), log(quantity+1), color = size1_descr))+
  geom_point()+
  geom_smooth()+
  theme_bw()+
  labs(title = 'Curva de demanda',
       y = 'Cantidad Demandada (# Helados)',
       x = 'Precio (USD)', color = 'Tamaño')
```



13. Grafica la curva de demanda por sabor. Crea una variable con los 3 sabores más populares y agruga el resto de los sabores como 'otros'. Parece haber diferencias en la elasticidad precio dependiendo del sabor?

```
# Detectando las top frequencies
freq_sabores<-
  base %>%
  group_by(flavor_descr) %>%
  tally() %>%
  arrange(desc(n))

base<-
  base %>%
  mutate(sabor = case_when(flavor_descr == freq_sabores$flavor_descr[1] ~ freq_sabores$flavor_descr[1],
                           flavor_descr == freq_sabores$flavor_descr[2] ~ freq_sabores$flavor_descr[2],
                           flavor_descr == freq_sabores$flavor_descr[3] ~ freq_sabores$flavor_descr[3]),
         sabor = replace_na(sabor, replace = 'Other'))

ggplot(base, aes(log(price_unit+1), log(quantity+1), color = sabor))+
  geom_point()+
  geom_smooth()+
  theme_bw()+
  labs(title = 'Curva de demanda',
       y = 'Cantidad Demandada (# Helados)',
       x = 'Precio (USD)', color = 'Tamaño')
```



Estimación

14. Estima la regresión de la curva de demanda de los helados. Reporta la tabla de la regresión

Algunos tips:

- No olvides borrar la variable que recien creamos de sabores. Incluirla (dado que es perfectamente colineal con flavor), sería una violación a supuesto GM 3 de la regresión.
- No olvides quitar `quantity`, `price_unit`, `price_deal` y otras variables que sirven como identificadora. Tambien quitar `fips_state_code` y `fips_county_code`.
- Empecemos con una regresión que incluya a todas las variables.

Nota: La regresión en `R` entiende que si le metes variables de texto, debe convertirlas a un factor. En algunos otros algoritmos que veremos durante el curso, tendremos que convertir manualmente toda la base a una númerica.

Quitemos las fechas

```
base$female_head_birth<-NULL
base$male_head_birth<-NULL
```

```
base<-
  base %>%
  mutate(log_quantity = log(quantity+1),
         log_price = log(price_unit+1))

base_estimacion<-
  base %>%
  ungroup() %>%
  select(-c(quantity, price_paid_deal, price_paid_non_deal, price, price_unit, sabor, primary_key, fips_
```

```r
var_na<-map_dbl(base_estimacion %>% select_all(),
              ~100*sum(is.na(.))/nrow(base))

var_na<-var_na[var_na>0]


fit <- lm(log_quantity ~ ., data = base_estimacion %>% select(-household_id, -fips_county_code))

resultados<-tidy(fit)
kable(tidy(fit))
```

term	estimate	std.error	statistic	p.value
(Intercept)	0.9716805	0.0410434	23.6744678	0.0000000
coupon_value	0.0392713	0.0033055	11.8807529	0.0000000
promotion_type2	-0.0220520	0.0085834	-2.5691354	0.0102019
promotion_type3	0.0291219	0.0083510	3.4872404	0.0004890
promotion_type4	0.0269453	0.0198199	1.3595096	0.1739992
promotion_typeno promotion	-0.0134056	0.0034847	-3.8469726	0.0001199
size1_descr32.0 MLOZ	-0.0213792	0.0123066	-1.7372196	0.0823626
flavor_descrBANANA SPLIT	0.0007276	0.0115282	0.0631145	0.9496759
flavor_descrBLACK & TAN	0.1130267	0.0439540	2.5714770	0.0101332
flavor_descrBROWNIE BATTER	-0.0293992	0.0193236	-1.5214143	0.1281704
flavor_descrBUTTER PECAN	0.0071044	0.0158088	0.4493926	0.6531530
flavor_descrCAKE BATTER	-0.0371244	0.0129854	-2.8589261	0.0042548
flavor_descrCHC	-0.0366042	0.0238575	-1.5342821	0.1249748
flavor_descrCHC ALMOND NOUGAT	-0.0286719	0.0212363	-1.3501336	0.1769872
flavor_descrCHC CHIP C-DH	-0.0320488	0.0100297	-3.1953865	0.0013984
flavor_descrCHC FUDGE BROWNIE	0.0078889	0.0096838	0.8146432	0.4152855
flavor_descrCHERRY GRCA	0.0037971	0.0088625	0.4284427	0.6683331
flavor_descrCHUBBY HUBBY	-0.0160685	0.0142232	-1.1297412	0.2585977
flavor_descrCHUNKY MONKEY	-0.0090163	0.0099198	-0.9089236	0.3634005
flavor_descrCINNAMON BUNS	0.0167041	0.0114401	1.4601327	0.1442680
flavor_descrCOFFEE	-0.0462161	0.0297635	-1.5527803	0.1204902
flavor_descrCREME BRULEE	0.0333653	0.0125865	2.6508704	0.0080343
flavor_descrDOUBLE CHC FUDGE SWR	-0.0984592	0.2159174	-0.4560039	0.6483917
flavor_descrDUBLIN MUDSLIDE	-0.0267305	0.0135068	-1.9790372	0.0478244
flavor_descrFOSSIL FUEL	0.0054645	0.0247087	0.2211568	0.8249724
flavor_descrHALF BAKED	-0.0199436	0.0109861	-1.8153456	0.0694846
flavor_descrHEATH CANDY EVERYTHING BUT THE	-0.0099700	0.0119637	-0.8333514	0.4046557
flavor_descrHEATH COFFEE CRUNCH	0.0154712	0.0099458	1.5555394	0.1198322
flavor_descrHEATH CRUNCH	-0.0205310	0.0122791	-1.6720254	0.0945337
flavor_descrIMAGINE WHIRLED PEACE	-0.0318677	0.0114046	-2.7942927	0.0052059
flavor_descrKARAMEL SUTRA	-0.0095093	0.0108442	-0.8769039	0.3805485
flavor_descrMAGIC BROWNIES	-0.0051142	0.0170453	-0.3000334	0.7641545
flavor_descrMINT CHC CHUNK	-0.0115267	0.0193863	-0.5945807	0.5521300
flavor_descrNEAPOLITAN DYNAMITE	-0.0420048	0.0173897	-2.4155074	0.0157215
flavor_descrNEW YORK SUPER FUDGE CHUNK	-0.0075059	0.0102365	-0.7332516	0.4634129
flavor_descrOATMEAL COOKIE CHUNK	-0.0376761	0.0176641	-2.1329150	0.0329429
flavor_descrONE CSK BROWNIE	-0.0522357	0.0117728	-4.4369857	0.0000092
flavor_descrOXFORD MINT CHC COOKIE	-0.0415069	0.0140670	-2.9506579	0.0031744
flavor_descrPB CUP	-0.0160932	0.0105231	-1.5293220	0.1261992

term	estimate	std.error	statistic	p.value
flavor_descrPB TRUFFLE	-0.1398441	0.2160743	-0.6472039	0.5175068
flavor_descrPHISH FOOD	-0.0103230	0.0106155	-0.9724508	0.3308371
flavor_descrPISTACHIO PISTACHIO	0.0139010	0.0109474	1.2698006	0.2041692
flavor_descrPUMPKIN CSK	0.0526129	0.0195012	2.6979263	0.0069827
flavor_descrRSP CHC CHUNK	-0.0275627	0.0291156	-0.9466631	0.3438210
flavor_descrSMORES	-0.0540472	0.0170552	-3.1689501	0.0015320
flavor_descrSTR	-0.0461794	0.0653016	-0.7071711	0.4794677
flavor_descrSTR CSK	-0.0311629	0.0120655	-2.5828138	0.0098063
flavor_descrSTRAWBERRIES & CREAM	-0.0201117	0.0617584	-0.3256513	0.7446913
flavor_descrSWEET CREAM & COOKIES	-0.0654182	0.0527932	-1.2391405	0.2153068
flavor_descrTRIPLE CARAMEL CHUNK	0.0098875	0.0243079	0.4067596	0.6841885
flavor_descrTURTLE SOUP	-0.0346930	0.0168906	-2.0539878	0.0399888
flavor_descrVAN	-0.0233483	0.0120942	-1.9305451	0.0535523
flavor_descrVAN CARAMEL FUDGE	0.0260149	0.0147540	1.7632431	0.0778735
flavor_descrVERMONTY PYTHON	-0.0314782	0.0201010	-1.5659983	0.1173635
flavor_descrW-N-C-P-C	-0.0253457	0.0110269	-2.2985306	0.0215410
flavor_descrWHITE RUSSIAN	-0.1497220	0.2149842	-0.6964328	0.4861653
formula_descrREGULAR	0.0040926	0.0143378	0.2854396	0.7753101
household_size	0.0144207	0.0023008	6.2676476	0.0000000
household_income	0.0000067	0.0003584	0.0188200	0.9849849
age_of_female_head	NA	NA	NA	NA
age_of_male_head	NA	NA	NA	NA
age_and_presence_of_children2	-0.0044310	0.0094774	-0.4675343	0.6401223
age_and_presence_of_children3	0.0073793	0.0085494	0.8631334	0.3880736
age_and_presence_of_children4	-0.0300954	0.0117451	-2.5623757	0.0104025
age_and_presence_of_children5	0.0441318	0.0246173	1.7927115	0.0730330
age_and_presence_of_children6	0.0206758	0.0107999	1.9144373	0.0555774
age_and_presence_of_children7	-0.0027849	0.0182812	-0.1523349	0.8789242
age_and_presence_of_children9	0.0407999	0.0078846	5.1746144	0.0000002
male_head_employment1	0.0065446	0.0301991	0.2167149	0.8284326
male_head_employment2	0.0132835	0.0308086	0.4311616	0.6663551
male_head_employment3	0.0216089	0.0292092	0.7397990	0.4594300
male_head_employment9	0.0386326	0.0295140	1.3089606	0.1905615
female_head_employment1	-0.0206456	0.0168807	-1.2230277	0.2213325
female_head_employment2	-0.0309659	0.0175072	-1.7687513	0.0769494
female_head_employment3	-0.0025191	0.0161344	-0.1561305	0.8759316
female_head_employment9	-0.0202878	0.0239756	-0.8461847	0.3974590
male_head_education	-0.0033622	0.0019331	-1.7392529	0.0820044
female_head_education	0.0019794	0.0018818	1.0518829	0.2928650
marital_status2	-0.0017933	0.0210629	-0.0851383	0.9321522
marital_status3	-0.0122562	0.0202241	-0.6060209	0.5445072
marital_status4	-0.0290290	0.0202497	-1.4335530	0.1517142
male_head_occupation2	0.0097698	0.0050242	1.9445646	0.0518403
male_head_occupation3	0.0165705	0.0075101	2.2064247	0.0273647
male_head_occupation4	-0.0161806	0.0072214	-2.2406448	0.0250591
male_head_occupation5	0.0213263	0.0063391	3.3642468	0.0007689
male_head_occupation6	0.0087272	0.0073125	1.1934702	0.2326983
male_head_occupation7	0.0399448	0.0201390	1.9834501	0.0473297
male_head_occupation8	-0.0198215	0.0079917	-2.4802517	0.0131365
male_head_occupation9	-0.0444709	0.0266269	-1.6701487	0.0949043
male_head_occupation10	-0.0285132	0.0199994	-1.4257044	0.1539680
male_head_occupation11	0.0102791	0.0141888	0.7244519	0.4687961

term	estimate	std.error	statistic	p.value
male__head__occupation12	-0.0027777	0.0083420	-0.3329775	0.7391544
female__head__occupation10	0.0355751	0.0247979	1.4346003	0.1514154
female__head__occupation11	0.0370500	0.0388833	0.9528509	0.3406762
female__head__occupation12	0.0454988	0.0180697	2.5179651	0.0118106
female__head__occupation2	0.0025392	0.0055714	0.4557600	0.6485671
female__head__occupation3	-0.0116296	0.0064476	-1.8036983	0.0712924
female__head__occupation4	0.0031534	0.0077531	0.4067359	0.6842060
female__head__occupation5	0.0019769	0.0144483	0.1368234	0.8911717
female__head__occupation6	-0.0291699	0.0125469	-2.3248774	0.0200877
female__head__occupation7	-0.0680119	0.0364467	-1.8660654	0.0620457
female__head__occupation8	0.0430157	0.0080208	5.3629971	0.0000001
female__head__occupation9	0.0659946	0.0476887	1.3838621	0.1664149
female__head__occupationOther	NA	NA	NA	NA
household__composition2	0.0196791	0.0210953	0.9328670	0.3508990
household__composition3	0.0072794	0.0221628	0.3284499	0.7425747
household__composition5	0.0433555	0.0208172	2.0826796	0.0372921
household__composition6	NA	NA	NA	NA
household__composition7	0.0124797	0.0250171	0.4988476	0.6178918
household__composition8	0.0144373	0.0215642	0.6695043	0.5031810
race2	0.0028083	0.0060793	0.4619510	0.6441211
race3	0.0037167	0.0093884	0.3958797	0.6921977
race4	-0.0063606	0.0085967	-0.7398948	0.4593718
hispanic__origin	-0.0006281	0.0079572	-0.0789310	0.9370883
region2	0.0022828	0.0101159	0.2256648	0.8214642
region3	-0.0073780	0.0098312	-0.7504677	0.4529812
region4	0.0314855	0.0109396	2.8781165	0.0040045
scantrack__market__identifier10	-0.0017911	0.0160255	-0.1117669	0.9110093
scantrack__market__identifier11	0.0674732	0.0183140	3.6842532	0.0002299
scantrack__market__identifier12	-0.0236733	0.0151792	-1.5595914	0.1188710
scantrack__market__identifier13	-0.0099079	0.0155738	-0.6361916	0.5246582
scantrack__market__identifier14	-0.0225973	0.0159311	-1.4184382	0.1560772
scantrack__market__identifier15	0.0011743	0.0219621	0.0534697	0.9573581
scantrack__market__identifier16	0.0247713	0.0166903	1.4841737	0.1377773
scantrack__market__identifier17	0.0120954	0.0159093	0.7602738	0.4470992
scantrack__market__identifier18	-0.0549667	0.0154563	-3.5562552	0.0003770
scantrack__market__identifier19	0.0015470	0.0153241	0.1009541	0.9195878
scantrack__market__identifier2	0.0471679	0.0153983	3.0631880	0.0021926
scantrack__market__identifier20	0.0521559	0.0160470	3.2501980	0.0011550
scantrack__market__identifier21	0.0867761	0.0223728	3.8786502	0.0001053
scantrack__market__identifier22	-0.0184268	0.0164911	-1.1173841	0.2638425
scantrack__market__identifier23	0.0970703	0.0202156	4.8017471	0.0000016
scantrack__market__identifier24	0.0310400	0.0116042	2.6748869	0.0074810
scantrack__market__identifier25	0.0029441	0.0142841	0.2061078	0.8367086
scantrack__market__identifier26	-0.0247526	0.0178201	-1.3890220	0.1648403
scantrack__market__identifier27	-0.0075494	0.0202073	-0.3735993	0.7087061
scantrack__market__identifier28	0.0043020	0.0166151	0.2589211	0.7956986
scantrack__market__identifier29	0.0235267	0.0168038	1.4000758	0.1615049
scantrack__market__identifier3	-0.0120832	0.0178569	-0.6766671	0.4986244
scantrack__market__identifier30	-0.0080087	0.0185128	-0.4326023	0.6653080
scantrack__market__identifier31	0.0335774	0.0243245	1.3803958	0.1674790
scantrack__market__identifier32	-0.0233756	0.0138727	-1.6850075	0.0920015
scantrack__market__identifier33	0.0894348	0.0129669	6.8971694	0.0000000

term	estimate	std.error	statistic	p.value
scantrack_market_identifier34	0.2499822	0.0231245	10.8102836	0.0000000
scantrack_market_identifier35	0.0215802	0.0249105	0.8663076	0.3863310
scantrack_market_identifier36	0.0607224	0.0221871	2.7368282	0.0062085
scantrack_market_identifier37	-0.0068954	0.0222015	-0.3105833	0.7561204
scantrack_market_identifier38	-0.0104313	0.0163814	-0.6367743	0.5242786
scantrack_market_identifier39	0.0064118	0.0174263	0.3679373	0.7129235
scantrack_market_identifier4	0.0304201	0.0198738	1.5306670	0.1258662
scantrack_market_identifier40	-0.0058719	0.0202156	-0.2904659	0.7714626
scantrack_market_identifier41	0.0345191	0.0166631	2.0715830	0.0383161
scantrack_market_identifier42	0.0030784	0.0151269	0.2035082	0.8387397
scantrack_market_identifier43	0.0218464	0.0167346	1.3054632	0.1917490
scantrack_market_identifier44	0.0327385	0.0164020	1.9960089	0.0459454
scantrack_market_identifier45	0.0163087	0.0278483	0.5856240	0.5581343
scantrack_market_identifier46	0.0105870	0.0183700	0.5763193	0.5644054
scantrack_market_identifier47	0.0363963	0.0234211	1.5539989	0.1201992
scantrack_market_identifier48	0.0368912	0.0284865	1.2950429	0.1953192
scantrack_market_identifier49	0.0054663	0.0182788	0.2990500	0.7649047
scantrack_market_identifier5	0.0061013	0.0231221	0.2638732	0.7918801
scantrack_market_identifier50	0.0433077	0.0164636	2.6305144	0.0085316
scantrack_market_identifier51	-0.0081582	0.0164014	-0.4974058	0.6189079
scantrack_market_identifier52	0.0192724	0.0172491	1.1172997	0.2638785
scantrack_market_identifier6	-0.0355340	0.0234997	-1.5121056	0.1305215
scantrack_market_identifier7	-0.0004552	0.0146677	-0.0310317	0.9752445
scantrack_market_identifier8	0.0379989	0.0116584	3.2593643	0.0011183
scantrack_market_identifier9	0.0266075	0.0144648	1.8394698	0.0658597
scantrack_market_identifierOther	0.0126050	0.0110690	1.1387574	0.2548169
type_of_residence2	-0.0487003	0.0112317	-4.3359665	0.0000146
type_of_residence3	-0.0004684	0.0089385	-0.0524071	0.9582048
type_of_residence4	-0.0531127	0.0225026	-2.3602963	0.0182691
type_of_residence5	-0.0204034	0.0050790	-4.0172127	0.0000591
type_of_residence6	-0.0215026	0.0073441	-2.9278780	0.0034164
type_of_residence7	-0.0072699	0.0080693	-0.9009370	0.3676318
kitchen_appliances	-0.0009317	0.0009177	-1.0152195	0.3100125
tv_items	0.0017809	0.0019222	0.9265143	0.3541890
household_internet_connection	0.0110396	0.0042424	2.6021858	0.0092694
log_price	-0.1898445	0.0100878	-18.8192887	0.0000000

15 (2 pts). Cuales son los elementos que guarda el objecto de la regresión? Listalos. Cual es el F-test de la regresión? Escribe la prueba de manera matemática (i.e. como la vimos en clase). (Tip: `summary(fit)` te arroja algo del F-test)

$$H_0 : \beta_i = 0 \quad H_a : Alguna \quad \beta_i \neq 0$$

$$F = \frac{ESS(n-k-1)}{RSS(k)} = \frac{R^2(n-k-1)}{(1-R^2)k} = \frac{0.08847(21940-174-1)}{(1-0.08847)174} = 12.14$$

$$p(F) = 4.004x \quad e^{-312} < 0.01$$

Por lo tanto, la regresión explica más que el modelo nulo.

```
a<-summary(fit)

a$fstatistic
```

```
##       value       numdf       dendf
##    12.14011   174.00000 21765.00000
```

```
pf(q = a$fstatistic[1], df1 = a$fstatistic[2], df2 = a$fstatistic[3], lower.tail = F)
```

```
##         value
## 4.004342e-312
```

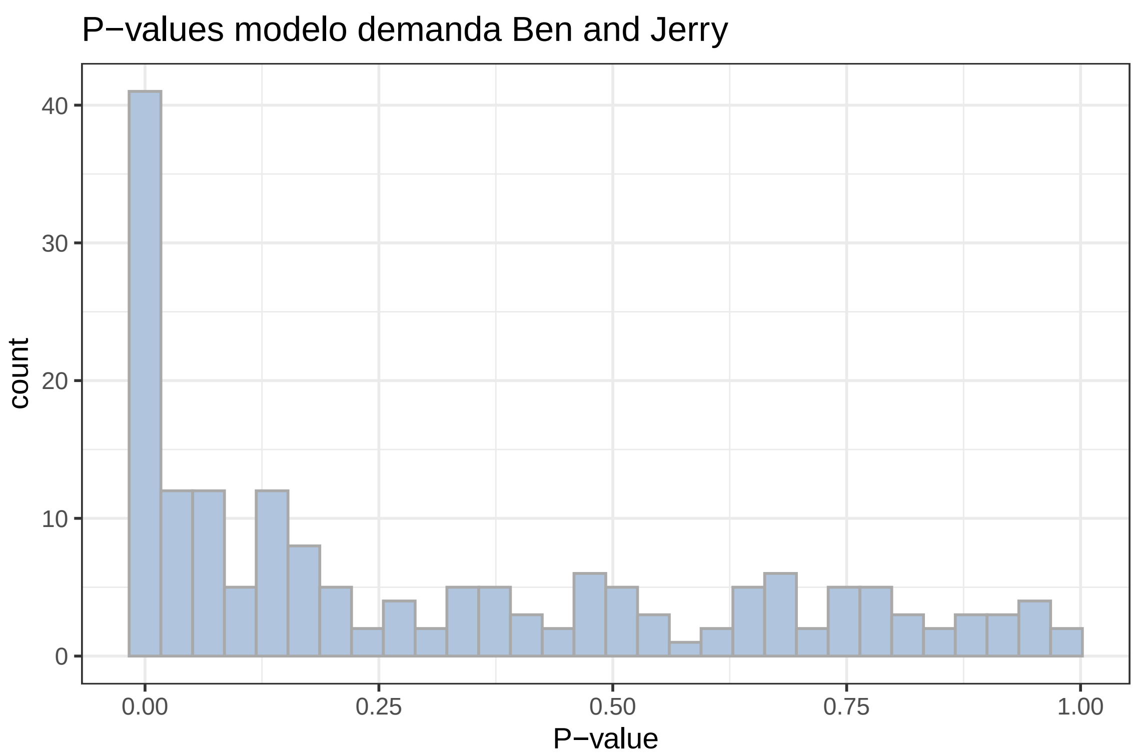
16. Cuál es la elasticidad precio de los helados Ben and Jerry ? Es significativo? Interpreta el coeficiente

$$\epsilon_p^Q = -0.1898^{***}$$

Esto se interpreta, si Ben and Jerry sube el precio de los helados 1 por ciento, la cantidad demandada caerá 0.1898 por ciento. Es un bien relativamente ineslástico.

17. Cuántos p-values tenemos en la regresión. Haz un histograma de los p-values.

```
ggplot(resultados, aes(p.value))+
  geom_histogram(fill = 'lightsteelblue', color = 'darkgrey')+
  theme_bw()+
  labs(title = 'P-values modelo demanda Ben and Jerry', x = 'P-value')
```



18 (4pts). Realiza un ajuste FDR a una $q = 0.10$. Grafica el procedimiento (con y sin zoom-in a p-values<0.05). Cuantas variables salían significativas con $\alpha = 0.05$? Cuantas salen con FDR?

Tip: crea el ranking de cada p-value como `resultados %>% arrange(p.value) %>% mutate(ranking = row_number)`

Con la inferencia clásica ($\alpha = 0.05$), salen 53 de 122 variables significativas.

Con FDR a $q = 0.1$, salen 45 variables significativas.

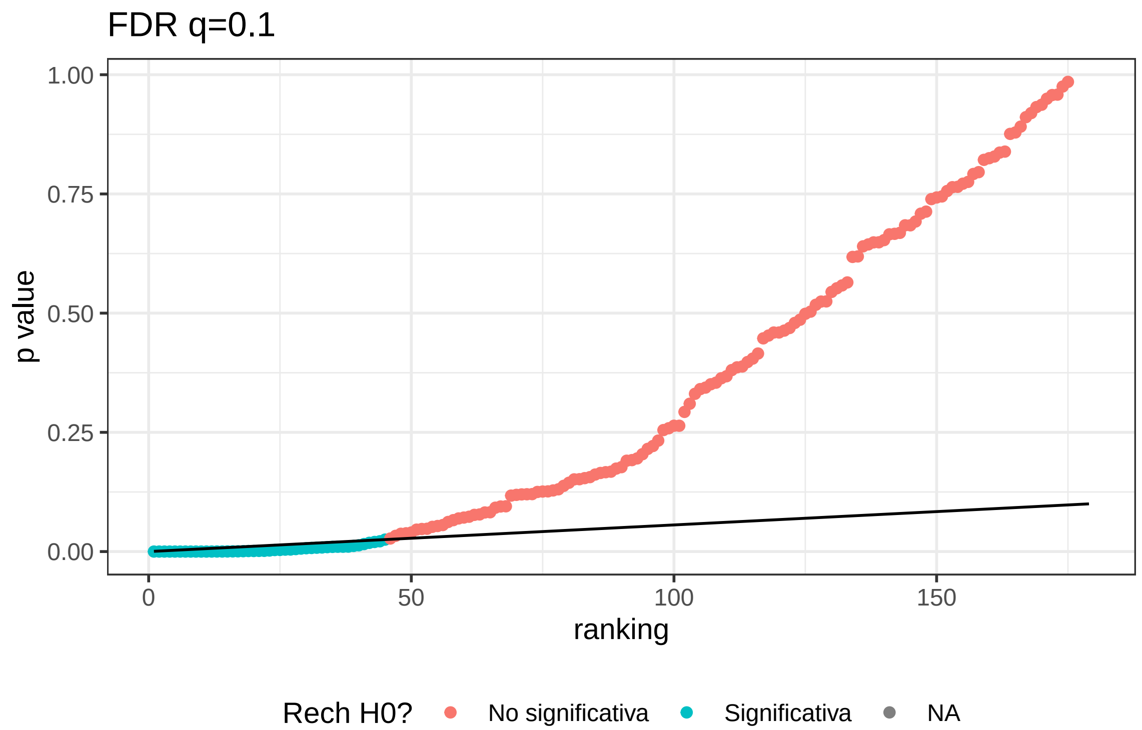
```r
# Cuantas salen con alpha 0.05

table(resultados$p.value<0.05)
```

```
##
## FALSE  TRUE
##   122    53
```

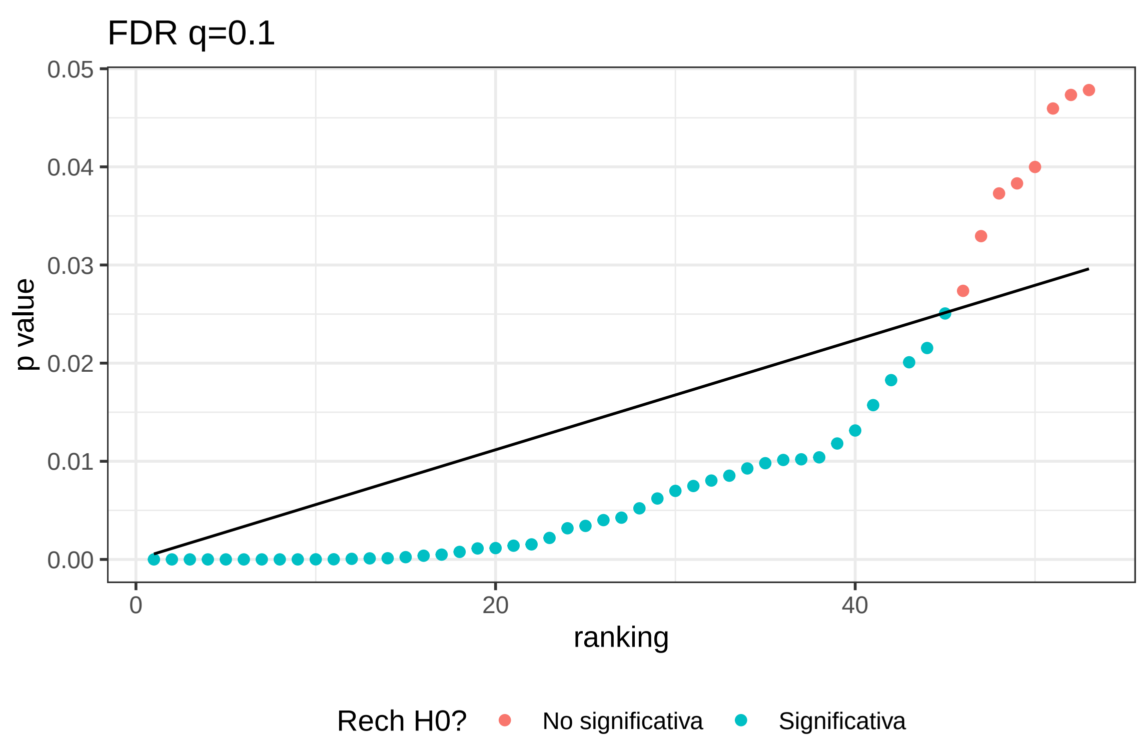
```r
# Creamos el ranking de los p-values
resultados<-
  resultados %>%
  arrange(p.value) %>%
  mutate(ranking = row_number())


resultados<-
  resultados %>%
  mutate(corte_fdr = 0.1*ranking/nrow(resultados),
         sig_fdr = if_else(p.value<=corte_fdr, 'Significativa', 'No significativa'))

table(resultados$sig_fdr)
```

```
##
## No significativa    Significativa
##              130               45
```

```r
# sin zoom -in
ggplot(resultados, aes(ranking, p.value, color = sig_fdr))+
  geom_point()+
  geom_line(aes(ranking, corte_fdr), color = 'black')+
  theme_bw()+
  theme(legend.position = 'bottom')+
  labs(title = 'FDR q=0.1', x = 'ranking', y = 'p value', color = 'Rech H0?')
```




```
# Con zoom -in
ggplot(resultados %>% filter(p.value<0.05), aes(ranking, p.value, color = sig_fdr))+
  geom_point()+
  geom_line(aes(ranking, corte_fdr), color = 'black')+
  theme_bw()+
  theme(legend.position = 'bottom')+
  labs(title = 'FDR q=0.1', x = 'ranking', y = 'p value', color = 'Rech H0?')
```



19 (2pts). Repite el ejercicio pero ahora con Holm-Bonferroni. Comparalo vs FDR. En este caso cuantas variables son significativas? Haz la grafica comparativa (solo con zoom-in)

En este caso tambien hay 45 significativas.

```
resultados <-
  resultados %>%
  mutate(corte_hb = 0.05/(nrow(resultados) + 1 - ranking),
         sig_hb = if_else(p.value<corte_fdr, 'Significativa', 'No Significativa'))


table(resultados$sig_hb)
```

```
##
## No Significativa    Significativa
##              130               45
```

```
resultados2<-
  resultados %>%
  pivot_longer(cols = c(corte_fdr, corte_hb), names_to = 'metodo', values_to = 'corte')

# Con zoom -in
ggplot(resultados2 %>% filter(p.value<0.05), aes(ranking, p.value, color = sig_hb, shape = sig_fdr))+
  geom_point()+
  geom_line(aes(ranking, corte, color = metodo))+
  theme_bw()+
  theme(legend.position = 'bottom')+
  labs(title = 'FDR q=0.1 vs Holm Bonferroni', x = 'ranking', y = 'p value', color = 'Rech H0?')
```