# EDS241: Assignment 4

Scout Leonard

03/08/2022

## 1 Read in Sardines Data

The code chunk below reads in the sardines data.

```
sardines <- read.csv(here("data","EU_sardines.csv"))
```

## 2 Homework Questions

### 2.1 a.) Estimate a bivariate regression of log(volume_sold_kg) on log(price euro_kg). What is the price elasticity of demand for sardines? Test the nullhypothesis that the price elasticity is equal to -1.

The code chunk below adds a variable to the `sardines` dataset that is the log of the price in Euros per kilogram of sardines.

```
#transform  variables to generate log values
sardines <- sardines %>%
  mutate(log_vol_sold_kg = log(volume_sold_kg),
         log_price_euro_kg = log(price_euro_kg))
```

The code chunk below runs a regression of `log_price_euro_kg` on `log_vol_sold_kg` and formats the output in a table.

```
vol_on_price <- lm_robust(log_vol_sold_kg ~ log_price_euro_kg, data = sardines)
```

```
vol_on_price_table <- tidy(vol_on_price)

vol_on_price_table %>%
  select(term, estimate, std.error, p.value, conf.low, conf.high) %>%
  kable()
```

| term | estimate | std.error | p.value | conf.low | conf.high |
|---|---|---|---|---|---|
| (Intercept) | 7.759061 | 0.0430246 | 0 | 7.674709 | 7.843413 |
| log_price_euro_kg | -1.545335 | 0.0781254 | 0 | -1.698505 | -1.392166 |

The code chunk below tests the null hypothesis that the price elasticity is equal to -1.

```
linearHypothesis(vol_on_price, c("log_price_euro_kg = -1"), white.adjust = "hc2")
```

```
## Linear hypothesis test
##
## Hypothesis:
## log_price_euro_kg = - 1
```

```
##
## Model 1: restricted model
## Model 2: log_vol_sold_kg ~ log_price_euro_kg
##
##   Res.Df Df  Chisq        Pr(>Chisq)
## 1   3987
## 2   3986  1 48.724 0.000000000002946 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the confidence interval for the linear regression, we can say with 95% confidence that the coefficient on volume sold in kilograms, price elasticity (beta 1) is contained by the range -1.69 and -1.39. Also, our linear hypothesis This means we can reject the null hypothesis that price elasticity is equal to -1.

## 2.2 b.) Like in Lecture 8 (see the IV.R script), we will use wind_m_s as an instrument for log(price_euro_kg). To begin, estimate the first-stage regression relating log(price_euro_kg) to wind_m_s. Interpret the estimated coefficient on wind speed. Does it have the expected sign? Also test for the relevance of the instrument and whether it is a "weak" instrument by reporting the proper F-statistic.

```
#first stage regression
wind_price_mod <- lm_robust(log_price_euro_kg ~ wind_m_s, data = sardines)
```

```
wind_price_table <- tidy(wind_price_mod)

wind_price_table %>%
  select(term, estimate, std.error, p.value, conf.low, conf.high) %>%
  kable()
```

| term | estimate | std.error | p.value | conf.low | conf.high |
|---|---|---|---|---|---|
| (Intercept) | -0.3048875 | 0.0273093 | 0 | -0.3584290 | -0.2513461 |
| wind_m_s | 0.0673459 | 0.0055995 | 0 | 0.0563677 | 0.0783240 |

The estimated coefficient on wind speed is 0.07, which tells us that for each 1 m/s increase in wind speed, the price of sardines in euros per kilogram increases by .7 euros.

I think that this does have the expected sign (positive) because I would expect demand for sardines to increase as wind speed increases which would negatively impacting supply. Wind speed does not impact demand for sardines, but the supply (ie demand is the same, but there are fewer sardines, and the price is driven up).

```
# F-test for non-weak and relevant instruments (Lecture 9, slides 13-14)
wind_hypothesis <- linearHypothesis(wind_price_mod,
                                    c("wind_m_s = 0"),
                                    white.adjust = "hc2")

wind_hypothesis_table <- tidy(wind_hypothesis)

wind_hypothesis_table %>%
  kable()
```

| res.df | df | statistic | p.value |
|---|---|---|---|
| 3987 | NA | NA | NA |
| 3986 | 1 | 144.6526 | 0 |

The F statistic is 144.65 in the case of this first stage regression. This means that the instrument is not weak because it is greater than 10.

## 2.3 c.) Estimate the TSLS estimator of the price elasticity of demand for sardines using wind_m_s as an instrument for log(price_euro_kg). What is the estimated price elasticity of demand for sardines?

```
tsls1 <- ivreg(log_vol_sold_kg ~ log_price_euro_kg | wind_m_s, data = sardines)
summary(tsls1)
```

```
##
## Call:
## ivreg(formula = log_vol_sold_kg ~ log_price_euro_kg | wind_m_s,
##     data = sardines)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8626 -1.9790 -0.2333  2.0950  6.2354
##
## Coefficients:
##                   Estimate Std. Error t value            Pr(>|t|)
## (Intercept)        7.75534    0.04331  179.08 <0.0000000000000002 ***
## log_price_euro_kg -1.08802    0.37003   -2.94              0.0033 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.728 on 3986 degrees of freedom
## Multiple R-Squared: 0.09529, Adjusted R-squared: 0.09506
## Wald test: 8.646 on 1 and 3986 DF,  p-value: 0.003297
```

The estimated price elasticity of demand for sardines is -1.09.

### 2.4 d.) Repeat the exercise in (c), but include fixed effects for each year, month, and country. [Hint: you can use the command "as.factor(country) + as.factor(year) +as.factor(month)" to the ivreg function in R]. Report the estimated price elasticity of demand and the F-statistic testing for relevant and non-weak instruments.

```
tsls2 <- ivreg(log_vol_sold_kg ~ log_price_euro_kg + as.factor(country) +
                 as.factor(year) + as.factor(month) | as.factor(country) +
                 as.factor(year) + as.factor(month) + wind_m_s, data = sardines)

summary(tsls2)
```

```
##
## Call:
## ivreg(formula = log_vol_sold_kg ~ log_price_euro_kg + as.factor(country) +
##     as.factor(year) + as.factor(month) | as.factor(country) +
##     as.factor(year) + as.factor(month) + wind_m_s, data = sardines)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.2940 -1.8317 -0.1353  1.9969  6.2894
##
## Coefficients:
##                                  Estimate Std. Error t value
## (Intercept)                       7.33742    0.20781  35.309
## log_price_euro_kg                -1.25004    0.46393  -2.694
## as.factor(country)Italy          -0.68925    0.12970  -5.314
## as.factor(country)Portugal        1.71563    0.34614   4.957
## as.factor(country)United Kingdom -0.07422    0.31428  -0.236
## as.factor(year)2014               0.14610    0.15281   0.956
## as.factor(year)2015               0.18487    0.15221   1.215
## as.factor(year)2016               0.21335    0.15320   1.393
## as.factor(year)2017               0.07400    0.15224   0.486
## as.factor(year)2018              -0.09137    0.15508  -0.589
## as.factor(year)2019               0.03602    0.19688   0.183
## as.factor(month)2                 0.06866    0.20972   0.327
## as.factor(month)3                 0.51583    0.20489   2.518
## as.factor(month)4                 0.91433    0.20297   4.505
## as.factor(month)5                 1.14887    0.20370   5.640
## as.factor(month)6                 1.14474    0.20164   5.677
## as.factor(month)7                 1.40047    0.21047   6.654
## as.factor(month)8                 1.26382    0.21692   5.826
## as.factor(month)9                 1.31072    0.21298   6.154
## as.factor(month)10                0.72059    0.22958   3.139
## as.factor(month)11                0.48128    0.22575   2.132
## as.factor(month)12                0.06683    0.21920   0.305
##                                            Pr(>|t|)
## (Intercept)                      < 0.0000000000000002 ***
## log_price_euro_kg                             0.00708 **
## as.factor(country)Italy              0.0000001129314 ***
## as.factor(country)Portugal           0.0000007476568 ***
## as.factor(country)United Kingdom             0.81332
## as.factor(year)2014                          0.33909
```

```
## as.factor(year)2015                          0.22461
## as.factor(year)2016                          0.16382
## as.factor(year)2017                          0.62692
## as.factor(year)2018                          0.55580
## as.factor(year)2019                          0.85483
## as.factor(month)2                            0.74339
## as.factor(month)3                            0.01185 *
## as.factor(month)4                   0.0000068372477 ***
## as.factor(month)5                   0.0000000181902 ***
## as.factor(month)6                   0.0000000146801 ***
## as.factor(month)7                   0.0000000000324 ***
## as.factor(month)8                   0.0000000061221 ***
## as.factor(month)9                   0.0000000008287 ***
## as.factor(month)10                           0.00171 **
## as.factor(month)11                           0.03308 *
## as.factor(month)12                           0.76049
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.648 on 3966 degrees of freedom
## Multiple R-Squared: 0.1522,  Adjusted R-squared: 0.1477
## Wald test: 9.963 on 21 and 3966 DF,  p-value: < 0.00000000000000022
```

The estimated price elasticity of demand is is -1.25.

```
fs2 <- lm_robust(log_price_euro_kg ~ wind_m_s + as.factor(country) + as.factor(year) + as.factor(month)

fs2_hypothesis <- linearHypothesis(fs2,
                                   c("wind_m_s = 0"),
                                   white.adjust = "hc2")

fs2_hypothesis_table <- tidy(fs2_hypothesis)

fs2_hypothesis_table %>%
  kable()
```

| res.df | df | statistic | p.value |
|---|---|---|---|
| 3967 | NA | NA | NA |
| 3966 | 1 | 77.65815 | 0 |

The f statistic for the instrument of wind speed is 77.66.