Introduction to Regression Analysis - Multivariate Regression

Preparing your workfile

We add the basic libraries needed for this week's work:

```
library(tidyverse)    # for almost all data handling tasks
library(ggplot2)      # to produce nice graphiscs
library(stargazer)    # to produce nice results tables
library(haven)        # to import stata file
library(AER)          # access to HS robust standard errors
source("stargazer_HC.r")  # includes the robust regression display
```

Introduction

The data are an extract from the Understanding Society Survey (formerly the British Household Survey Panel).

Data Upload - and understanding data structure

Upload the data, which are saved in a STATA datafile (extension `.dta`). There is a function which loads STATA file. It is called `read_dta` and is supplied by the `haven` package.

```
data_USoc <- read_dta("20222_USoc_extract.dta")
data_USoc <- as.data.frame(data_USoc)    # ensure data frame structure
names(data_USoc)
```

```
##  [1] "pidp"    "age"     "jbhrs"   "paygu"   "wave"    "cpi"     "year"
##  [8] "region"  "urate"   "male"    "race"    "educ"    "degree"  "mfsize9"
```

Let us ensure that categorical variables are stored as `factor` variables. It is easiest to work with these in R.

```
data_USoc$region <- as_factor(data_USoc$region)
data_USoc$male <- as_factor(data_USoc$male)
data_USoc$degree <- as_factor(data_USoc$degree)
data_USoc$race <- as_factor(data_USoc$race)
```

The pay information (`paygu`) is provided as a measure of the (usual) gross pay per month. As workers work for varying numbers of hours per week (`jbhrs`) we divide the monthly pay by the approximate monthly hours (`4*jbhrs`). We shall also adjust for increasing price levels (as measured by `cpi`). These two adjustments leave us with an inflation adjusted hourly wage. We call this variable `hrpay` and also calculate the natural log of this variable (`lnhrpay`).

```
data_USoc <- data_USoc %>%
              mutate(hrpay = paygu/(jbhrs*4)/(cpi/100)) %>%
              mutate(lnhrpay = log(hrpay))
```

As we wanted to save these additional variables we assign the result of the operation to `data_USoc`.

In this lecture we will use the firmsize variable `mfsize9`. When we import the data from the STATA dta file, the variables inherit certain attributes. You can see them if you look at the structure of the dataframe (`str(data_USoc)`). One of the attributes is the label and it contains a little bit of information on the variable.

```
attr(data_USoc$mfsize9,"label")
```

```
## [1] "firm size dummies (bin means)"
```

And we can also look at the unique values this variable takes

```
unique(data_USoc$mfsize9)
```

```
##  [1]   75   NA  350  150   17   37 1500  750    1    6
```

You can see that there are 9 possible values (which is where the 9 in the name comes from). Each respondent records the size of the firm they work for, but we do not observe the actual size. There are buckets or bins for firm sizes, e.g. 51 to 100, 101 to 200, 201 to 500, etc. And the number we see is the mid-value of the respective bucket into which a firm falls.

```
data_USoc %>% count(mfsize9)
```

```
## # A tibble: 10 x 2
##    mfsize9     n
##      <dbl> <int>
##  1       1  2269
##  2       6  7722
##  3      17  9599
##  4      37  9095
##  5      75  6766
##  6     150  5814
##  7     350  6788
##  8     750  3768
##  9    1500  7168
## 10      NA 74283
```

You can see that each of the 9 categories has a significant number of observations. We shall change the name of this variable to `fsize` to make it slightly more intuitive:

```
names(data_USoc)[names(data_USoc) == "mfsize9"] <- "fsize"
```

Before we continue, it is important to understand what type of variable this is. Let's use the `str` (structure) function.

```
str(data_USoc$fsize)
```

```
##  num [1:133272] 75 NA NA NA NA NA NA NA NA NA ...
##  - attr(*, "label")= chr "firm size dummies (bin means)"
##  - attr(*, "format.stata")= chr "%9.0g"
```

You can see that this is a numeric variable.

Initially we will use the variable, however, not as a numeric, but rather as a categorical variable. So we are not really interested the numerical differences between the bin means. This means that we should change the variable type to a factor variable. We define a new variable for this reason, `fsize_f`.

```
data_USoc$fsize_f <- as.factor(data_USoc$fsize)
str(data_USoc$fsize_f)
```

```
##  Factor w/ 9 levels "1","6","17","37",..: 5 NA NA NA NA NA NA NA NA NA ...
```

Looking again at `str(data_USoc$fsize_f)` we confirm that this is now a factor variable.

Cleaning the dataset

Now we will remove data from the dataframe which have missing observations in either the `fsize` or the `lnhrpay` variable. As we will not require these observations in this lecture we will remove these observations from `data_USoc`.

```
data_USoc <- data_USoc %>% filter(!is.na(lnhrpay)) %>%
                            filter(!is.na(fsize))
```

Let's produce some summary statistics. There are a number of ways to achieve this

```
summary(data_USoc$lnhrpay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -7.816   1.890   2.258   2.284   2.676   8.868
```

```
summary(data_USoc$fsize)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    17.0    75.0   303.5   350.0  1500.0
```

We can also use the stargazer function (after selecting the two variables we are interested in)

```
data_USoc %>% select(lnhrpay,fsize) %>% stargazer(type="text")
```

```
##
## ================================================================
## Statistic   N     Mean   St. Dev.  Min   Pctl(25) Pctl(75)  Max
## ----------------------------------------------------------------
## lnhrpay   58,789  2.284   0.631   -7.816  1.890    2.676   8.868
## fsize     58,789 303.522 484.630    1       17      350    1,500
## ----------------------------------------------------------------
```

Let's also calculate the average hourly log pay by firm size.

```
data_USoc %>% group_by(fsize) %>%
              summarise(mean_lnhrpay = mean(lnhrpay), sd_lnhrpay = sd(lnhrpay), n = n())
```

```
## # A tibble: 9 x 4
##   fsize mean_lnhrpay sd_lnhrpay     n
##   <dbl>        <dbl>      <dbl> <int>
## 1     1         2.07      0.871  2235
## 2     6         2.04      0.590  7695
## 3    17         2.14      0.597  9575
## 4    37         2.21      0.612  9048
## 5    75         2.31      0.587  6746
## 6   150         2.37      0.600  5796
## 7   350         2.39      0.551  6782
## 8   750         2.48      0.568  3759
## 9  1500         2.61      0.615  7153
```

You can see that the average pay increases with firm size.

Estimating the model

Now we will investigate whether the hourly pay correlates with the firmsize. Initially we will use the factor (categorical) version of the variable, `fsize_f`.

```r
mod1 <- lm(lnhrpay~fsize_f, data = data_USoc)
stargazer_HC(mod1)
```

```
##
## ===========================================================
##                                Dependent variable:
##                     ---------------------------------------
##                                      lnhrpay
## -----------------------------------------------------------
## fsize_f6                            -0.031**
##                                      (0.015)
##
## fsize_f17                            0.076***
##                                      (0.014)
##
## fsize_f37                            0.139***
##                                      (0.014)
##
## fsize_f75                            0.239***
##                                      (0.015)
##
## fsize_f150                           0.306***
##                                      (0.015)
##
## fsize_f350                           0.319***
##                                      (0.015)
##
## fsize_f750                           0.411***
##                                      (0.016)
##
## fsize_f1500                          0.548***
##                                      (0.015)
##
## Constant                             2.067***
##                                      (0.013)
##
## -----------------------------------------------------------
## Observations                          58,789
## R2                                    0.080
## Adjusted R2                           0.080
## Residual Std. Error          0.606 (df = 58780)
## F Statistic              636.491*** (df = 8; 58780)
## ===========================================================
## Note:                          *p<0.1; **p<0.05; ***p<0.01
##                     Robust standard errors in parenthesis
```

You can see that all firm size categories have been included as dummy variables except for the smallest. This is the base category and the estimated constant term is the average value for `lnhrpay` in that base category (compare to the above table). The other coefficients are the difference in mean pay relative to the base category. As you can see the average pay in all but one of the other categories is larger than in the smallest firms.

The smallest firm category was the base category as R orders factors alphabetically/numerically and "1" comes before all of the other category labels. You may have reasons to want to re-estimate the model with a different base category. Say you want the largest category to be the base category, as in the lecture slides, then we need to change the ordering of our levels in the factor variable `fsize`. (I had to google "r factor change level order" to remind myself of how to do that.)

```r
data_USoc$fsize_f <- relevel(data_USoc$fsize_f, "1500")  # sets 1500 as the first level
str(data_USoc$fsize_f)
```

```
##  Factor w/ 9 levels "1500","1","6",..: 6 8 8 8 8 8 8 8 7 7 ...
```

If we now re-estimate the above model we obtain different parameters, but in essence we are still estimating the same model and indeed the group means from the earlier table. The lecture slides and background notes explain in more detail.

```r
mod2 <- lm(lnhrpay~fsize_f, data = data_USoc)
stargazer_HC(mod2)
```

```
## 
## ===========================================================
##                              Dependent variable:           
##                      --------------------------------------
##                                     lnhrpay                
## -----------------------------------------------------------
## fsize_f1                           -0.548***               
##                                     (0.015)                
##                                                            
## fsize_f6                           -0.579***               
##                                     (0.010)                
##                                                            
## fsize_f17                          -0.472***               
##                                     (0.009)                
##                                                            
## fsize_f37                          -0.409***               
##                                     (0.010)                
##                                                            
## fsize_f75                          -0.310***               
##                                     (0.010)                
##                                                            
## fsize_f150                         -0.242***               
##                                     (0.011)                
##                                                            
## fsize_f350                         -0.229***               
##                                     (0.010)                
##                                                            
## fsize_f750                         -0.138***               
##                                     (0.012)                
##                                                            
## Constant                           2.615***                
##                                     (0.007)                
##                                                            
## -----------------------------------------------------------
## Observations                        58,789                 
## R2                                   0.080                 
## Adjusted R2                          0.080                 
## Residual Std. Error          0.606 (df = 58780)            
## F Statistic              636.491*** (df = 8; 58780)        
## ===========================================================
## Note:                           *p<0.1; **p<0.05; ***p<0.01
##                     Robust standard errors in parenthesis
```

In fact, you could estimate the model without the constant but a full set of variables for all the 9 categories. You need to tell R to not include a constant. You do that by adding `-1` to the model specification.

```
mod3 <- lm(lnhrpay~fsize_f-1, data = data_USoc)
stargazer_HC(mod3)
```

```
##
## ==========================================================
##                                Dependent variable:
##                     --------------------------------------
##                                     lnhrpay
## ----------------------------------------------------------
## fsize_f1500                        2.615***
##                                     (0.007)
##
## fsize_f1                           2.067***
##                                     (0.013)
##
## fsize_f6                           2.036***
##                                     (0.007)
##
## fsize_f17                          2.143***
##                                     (0.006)
##
## fsize_f37                          2.205***
##                                     (0.006)
##
## fsize_f75                          2.305***
##                                     (0.007)
##
## fsize_f150                         2.372***
##                                     (0.008)
##
## fsize_f350                         2.386***
##                                     (0.007)
##
## fsize_f750                         2.477***
##                                     (0.010)
##
## ----------------------------------------------------------
## Observations                        58,789
## R2                                   0.935
## Adjusted R2                          0.935
## Residual Std. Error          0.606 (df = 58780)
## F Statistic             93,462.350*** (df = 9; 58780)
## ==========================================================
## Note:                          *p<0.1; **p<0.05; ***p<0.01
##                     Robust standard errors in parenthesis
```

In the absence of a constant each coefficient replicates the sample mean of the respective firm size categories.

All the above specifications essentially estimate the same model.

Recall that we changed the firm size variable into a factor variable which allowed the use of the firm size categories. What would happen if used the variable `fsize`, but as the log firmsize, `log(fsize)`. Recall that this is a numerical variable, so R uses the actual numbers reported.

```r
mod4 <- lm(lnhrpay~log(fsize), data = data_USoc)
stargazer_HC(mod4)
```

```
##
## ===========================================================
##                                Dependent variable:
##                      --------------------------------------
##                                      lnhrpay
## -----------------------------------------------------------
## log(fsize)                          0.090***
##                                      (0.001)
##
## Constant                            1.905***
##                                      (0.006)
##
## -----------------------------------------------------------
## Observations                         58,789
## R2                                    0.075
## Adjusted R2                           0.075
## Residual Std. Error           0.607 (df = 58787)
## F Statistic              4,763.519*** (df = 1; 58787)
## ===========================================================
## Note:                          *p<0.1; **p<0.05; ***p<0.01
##                      Robust standard errors in parenthesis
```

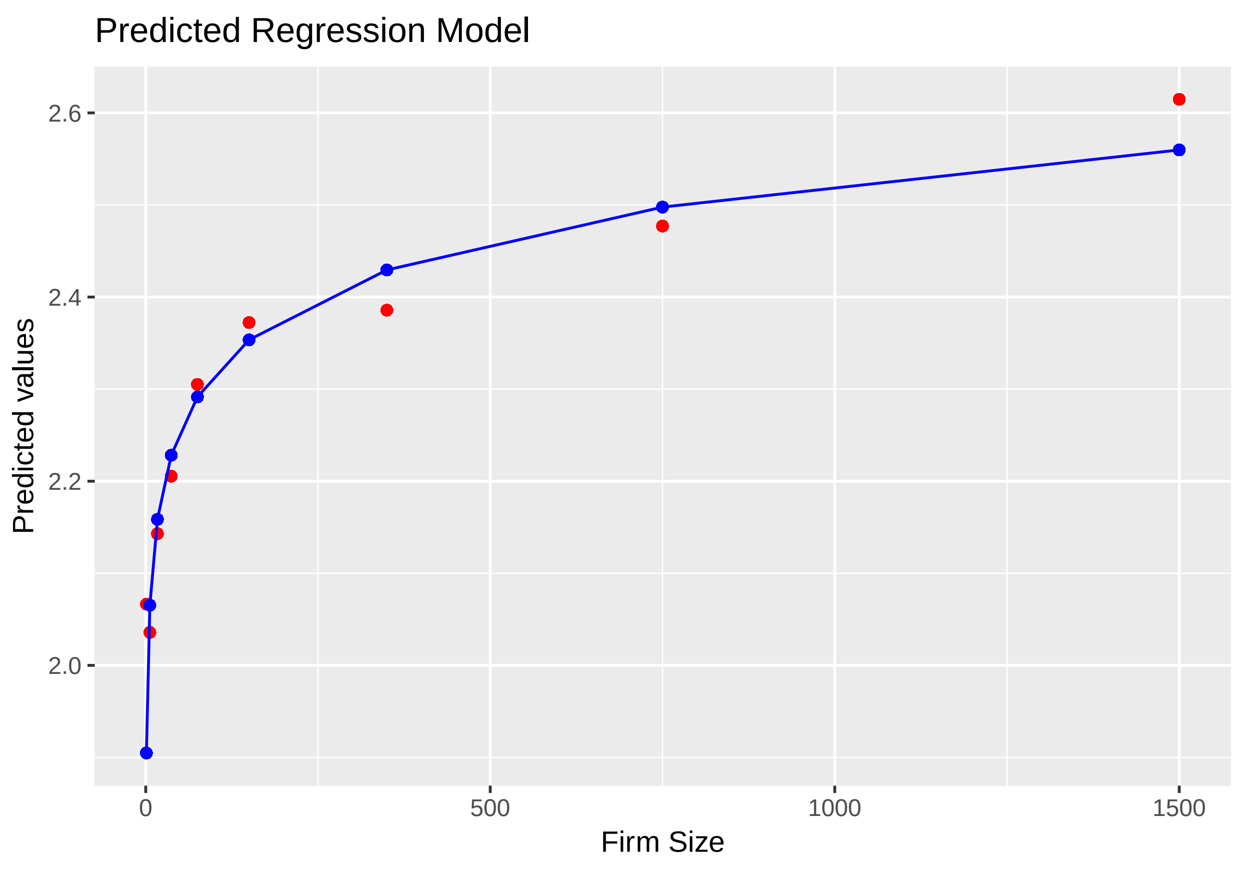
We see that estimated coefficient is positive (larger firms means larger wages) and statistically significant.

Let's compare the fitted values for `mod3` and `mod4`. First we add the predicted values to the dataframe

```r
data_USoc$pred_mod3 <- mod3$fitted.values
data_USoc$pred_mod4 <- mod4$fitted.values
```

Now we plot the predicted values for the two specifications. Recall that we only have 9 different values for the firm size.

```r
# pdf("Lecture5plot_R.pdf",width = 5.5, height = 4) # uncomment to save as pdf
ggplot(data_USoc, aes(x=fsize,y=pred_mod3)) +
  geom_point(color = "red") +
  geom_point(aes(y=pred_mod4),color = "blue") +
  geom_line(aes(y=pred_mod4),color = "blue") +
  ggtitle("Predicted Regression Model") +
  ylab("Predicted values") +
  xlab("Firm Size")
```



```
# dev.off() # uncomment to save as pdf
```