L.8: Multiple Regression

Econometrics 1: ver. 2024 Fall Semester

Naoki Awaya

Mutiple Regression Analysis

Multiple Regression Analysis

Simple linear regression model

- Linear regression model with a single explanatory variable:

$$Y = \beta_0 + X\beta_1 + \varepsilon$$

Multiple linear regression model

- Linear regression model with multiple explanatory variables:

$$Y = \beta_0 + X_1\beta_1 + \cdots + X_k\beta_k + \varepsilon$$

- If we use a vector notation, we can write simply as

$$Y = \mathbf{X}^\top \beta + \varepsilon,$$

where $\mathbf{X} = (1, X_1, \ldots, X_k)^\top$, and $\beta = (\beta_0, \beta_1, \ldots, \beta_k)^\top$.

Multiple Regression Analysis

- A linear regression model of annual income:

$$\text{Income} = \beta_0 + \text{Experience}\beta_1 + \text{Hours}\beta_2 + \text{Education}\beta_3 + \varepsilon$$

- For example, coefficient β_1 tells us

how much an additional year of working experience increases income

with other explanatory variables fixed (ignoring the correlation of Experience and the other variables).
$=$ the *ceteris paribus* effect of Experience

Multiple Regression Analysis

- Similar to the simple regression case, we can estimate $(\beta_0, \beta_1, \ldots, \beta_k)$ using the least squares method.
- That is, the OLS estimator of $(\beta_0, \beta_1, \ldots, \beta_k)$ can be obtained by solving[1]

$$\min_{(b_0, b_1, \ldots, b_k)} \frac{1}{n} \sum_{i=1}^{n} (Y_i - b_0 - X_{i1} b_1 - \cdots - X_{ik} b_k)^2$$

- We can show that the OLS estimator is unbiased, consistent, and normally distributed (for large n) under similar conditions as those given in Lecture 6.

[1] Without matrix-vector notation, deriving the exact form of the OLS estimator for multiple regression is extremely cumbersome.

Implementation

- Again, we use the R built-in data, `state.x77`.

```
>   library(janitor)
>   library(tidyverse)
>   data <- clean_names(as.data.frame(state.x77))
>   head(data, 4)
```

```
##            population income illiteracy life_exp murder hs_grad frost    ar
## Alabama          3615   3624        2.1    69.05   15.1    41.3    20   507
## Alaska            365   6315        1.5    69.31   11.3    66.7   152 5664
## Arizona          2212   4530        1.8    70.55    7.8    58.1    15 1134
## Arkansas         2110   3378        1.9    70.66   10.1    39.9    65   519
```

```
>   dim(data)
```

```
## [1] 50  8
```

`clean_names()`: remove blanks in variable names.

Implementation

- Suppose we would like to know the impacts of `illiteracy` (illiteracy rate) and `frost` (# days with minimum temperature below freezing) on `income` (per capita income).
- We consider the following multiple regression model:

$$\texttt{income} = \beta_0 + \texttt{illiteracy}\beta_1 + \texttt{frost}\beta_2 + \varepsilon$$

- We can easily estimate this model by using the `lm()` function:

$$\texttt{reg <- lm(income} \sim \underbrace{\texttt{illiteracy + frost}}_{\text{the model}}\texttt{,}\ \underbrace{\texttt{data}}_{\text{the data used}}\texttt{)}$$

- An intercept term (β_0) is automatically included.
- To see the estimation summary, use the `summary()` function.

Implementation

```
>   reg <- lm(income ~ illiteracy + frost, data)
>   summary(reg)
```

```
##
## Call:
## lm(formula = income ~ illiteracy + frost, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -905.88 -356.36  -60.19  293.55 2121.14
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5200.485    397.071  13.097  < 2e-16 ***
## illiteracy  -523.866    177.665  -2.949  0.00496 **
## frost         -1.453      2.083  -0.697  0.48902
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 561.4 on 47 degrees of freedom
## Multiple R-squared:  0.1993, Adjusted R-squared:  0.1652
## F-statistic:  5.85 on 2 and 47 DF,  p-value: 0.005386
```

Implementation

```
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5200.4846   397.0714 13.0971    0.000
## illiteracy  -523.8665   177.6646 -2.9486    0.005
## frost         -1.4528     2.0833 -0.6974    0.489
```

- `Estimate`: coefficient estimates
- `Std. Error`: standard errors
- `t value`: t-values (= `Estimate`/`Std. Error`)
- `Pr(>|t|)`: p-values

Implementation

Summary of the results

- `illiteracy` is statistically significantly negative at the 1% level.
- 1 percent increase in the illiteracy rate decreases the average income by about 524 USD.
- `frost` has a negative impact on the income level but is insignificant.
- `R-squared` (the *coefficient of determination*): how much the total variation of Y can be explained by the included X's.
    - In the above multiple regression model, `illiteracy` and `frost` contribute about 20% of the total variation of `income`.
    - **IMPORTANT**: a high R-squared does not necessarily mean that the model is a good model.

Multicollinearity

Multicollinearity

- Suppose that you have two explanatory variables X_1 and X_2, where X_2 is some linear transformation of X_1.[2]
- In this case, X_1 and X_2 are perfectly correlated:

```
> n <- 100
> X1 <- rnorm(n)
> X2 <- 2 - 3*X1
> cor(X1,X2)
```

```
## [1] -1
```

[2]That is, there are a and b satisfying $X_1 = a + bX_2$. An extreme case is $X_1 = X_2$.

Multicollinearity

- When X_1 and X_2 are perfectly correlated, we cannot include both in a regression model:

```
>  Y <- 1 + X1 + X2 + rnorm(n)
>  lm(Y ~ X1 + X2)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2)
##
## Coefficients:
## (Intercept)           X1           X2
##       3.187       -2.200           NA
```

- `NA` means "not available".
- This problem occurs because X_1 and X_2 are essentially the same variables and their impacts cannot be uniquely disentangled.[3]

[3] In the words of linear algebra, the regressors are linearly dependent.

Multicollinearity

- Even when X_1 and X_2 are not perfectly correlated, if they are "highly" correlated, the estimates entail large errors:

```
>  X1 <- rnorm(n)
>  X2 <- X1 + rnorm(n)/40
>  cor(X1,X2)
```

```
## [1] 0.99972
```

```
>  Y <- 1 + X1 + X2 + rnorm(n) # The true coef of (X1, X2) are (1,1)
>  lm(Y ~ X1 + X2)
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2)
##
## Coefficients:
## (Intercept)           X1           X2
##       1.172       -1.104        3.090
```

Multicollinearity

- This problem is known as multicollinearity.
- In the presence of multicollinearity, the estimated regression coefficients have large variances, and we cannot make precise statistical inference.
- To avoid multicollinearity, it is always better to check in advance if the correlations of the regressors are not too strong (say, $-0.8 \sim 0.8$).

Importing External Data

Importing CSV files into R

- In order to perform statistical analysis using external dataset, you need to import the data into **R**.
- **R** can read many different data file formats, including
  - csv (Comma-Separated Values) file
  - text file
  - Excel files (the package **xlsx** is needed)
  - etc
- For compatibility with other softwares and ease of editing, csv is the most commonly used format.

Importing CSV files into R

- The working directory is the folder where **R** will look for data files and save output files.
- The current working directory can be identified using the `getwd()` command.

```
>   getwd()
```

```
## [1] "C:/Users/naway/Dropbox/lecture_materials/Econometrics 1 (EDI
```

- The working directory can be changed using the `setwd()` command:[4]

```
setwd("location of the new directory")
```

[4] Setting working directory can be done manually through the menu bar: [File] → [Change dir...]. If you are an R-studio user, [Session] → [Set Working Directory] → [Choose Directory...]

Importing CSV files into R

- Once the working directory is set, csv files in the working directory can be read by the `read.csv()` command:

```
read.csv("name of the csv file")
```

- If you type just `read.csv("XXX.csv")`, you can only view the data content of the csv file.
- To perform statistical analysis on the imported data, you need to store the data in **R**.

```
data <- read.csv("name of the csv file")
```

Apartment Prices in Tokyo

- Practice data set: **apartments.csv**
  - Data on individual apartment transactions within Tokyo's 23 wards.
- The data csv file is available on **Waseda Moodle**.
- Set your working directory, and import the csv file by `read.csv()`:

```
>   data <- read.csv("apartments.csv")
>   head(data, 4)
```

```
##    price  area floor renov    stdist com ind
## 1 20.038 19.70     3     1 0.3123682   1   0
## 2 96.300 91.24    23     0 0.3116436   0   1
## 3 39.300 42.08    13     0 0.2460939   1   0
## 4 85.600 74.36    15     0 0.4952629   0   0
```

```
>   dim(data)
```

```
## [1] 500   7
```

Apartment Prices in Tokyo

Dependent variable (1st column)

price Price of the property (million JPY)

Explanatory variables (2nd - 7th columns)

area Area of the property (m^2)

floor Floor level of the property.

renov Dummy variable: 1 when the property has a history of renovations; 0 otherwise.

stdist Distance (km) to the nearest railway station.

com Dummy variable: 1 when the property is located in a commercially zoned area; 0 otherwise.

ind Dummy variable: 1 when the property is located in an industrially zoned area; 0 otherwise.

Apartment Prices in Tokyo

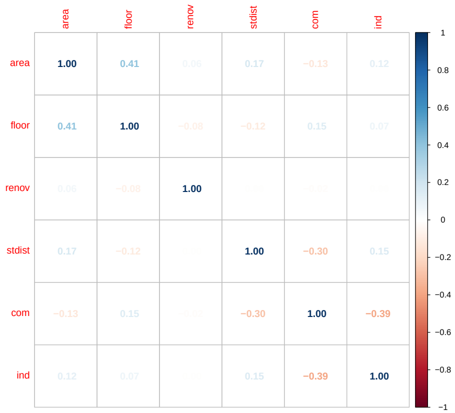
- Check if any high correlations exist between the explanatory variables.

```
>    cor(data[,-1]) %>% round(3)
```

```
##           area  floor  renov stdist    com    ind
## area     1.000  0.408  0.057  0.171 -0.134  0.120
## floor    0.408  1.000 -0.080 -0.117  0.148  0.073
## renov    0.057 -0.080  1.000 -0.005 -0.017 -0.005
## stdist   0.171 -0.117 -0.005  1.000 -0.295  0.153
## com     -0.134  0.148 -0.017 -0.295  1.000 -0.390
## ind      0.120  0.073 -0.005  0.153 -0.390  1.000
```

Apartment Prices in Tokyo

```
> library(corrplot)
> corrplot(cor(data[,-1]), method = "number")
```



Apartment Prices in Tokyo

- We estimate the following multiple regression model:

$$\begin{aligned}\text{price} = \beta_0 + \beta_1\text{area} + \beta_2\text{floor} + \beta_3\text{renov} + \beta_4\text{stdist}\\ + \beta_5\text{com} + \beta_6\text{ind} + \varepsilon.\end{aligned}$$

- We use the `lm()` function:

```
reg <- lm(price ~ area + floor + renov + stdist
              + com + ind, data)
```

or equivalently,[5]

```
reg <- lm(price ~ ., data)
```

[5] This shorthand can be used when the variables except for the one specified as the dependent variable are all used as the regressors.

Apartment Prices in Tokyo

```r
>   reg <- lm(price ~ ., data)
>   summary(reg)$coefficients %>% round(4)
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.2350     1.7873  2.9289   0.0036
## area          0.5841     0.0276 21.1998   0.0000
## floor         1.0954     0.0852 12.8627   0.0000
## renov        -6.0635     1.4695 -4.1263   0.0000
## stdist       -9.6555     2.2693 -4.2548   0.0000
## com          -2.4180     1.3855 -1.7452   0.0816
## ind          -3.9995     1.5777 -2.5350   0.0116
```

Apartment Prices in Tokyo

Summary of the results

- All explanatory variables except `com` are statistically significant at less than 5% level.
- `com` is significant at the 10% level.
- 1 m^2 increase in area size increases the property price by about 600,000 JPY.
- 1 km increase in distance to railway station decreases the property price by about 10 mil. JPY.
  - Too huge? probably because the data are only those in Tokyo Metropolitan district.

Apartment Prices in Tokyo

- You can also easily compute the confidence interval for each coefficient using the `confint()` command.

```
>   confint(reg, level = 0.95) %>% round(4)
```

```
##                   2.5 %  97.5 %
## (Intercept)    1.7233  8.7467
## area           0.5299  0.6382
## floor          0.9281  1.2628
## renov         -8.9506 -3.1763
## stdist       -14.1143 -5.1968
## com           -5.1401  0.3042
## ind           -7.0993 -0.8997
```

- The interval includes zero $\iff$ not significantly different from zero at the 5% level.

Transformation of Variables

Transformation of Y

Change the scale/unit of Y

- In the above apartment data, one may measure the apartment price in thousand JPY, for example, instead of million JPY.
- The new dependent variable is $Y^* = 1000Y$.
- When the dependent variable is multiplied by a constant c, then the resulting coefficient estimates will be also multiplied by c:

$$
\begin{aligned}
Y &= \beta_0 + X\beta_1 + \epsilon \\
Y^* = cY &= c\beta_0 + X{\color{red}c\beta_1} + c\epsilon
\end{aligned}
$$

- You should choose an appropriate scale (i.e., c) to avoid too large/small coefficient values.

Transformation of Y

```
>   lm(price ~ ., data)$coef %>% round(4)
```

```
## (Intercept)        area       floor       renov      stdist         com
##      5.2350      0.5841      1.0954     -6.0635     -9.6555     -2.4180
##         ind
##     -3.9995
```

```
>   data$price1 <- data$price*1000 # 1000 JPY
>   lm(price1 ~ area + floor + renov + stdist + com + ind, data)$coef %>%
+     round(4)
```

```
## (Intercept)        area       floor       renov      stdist         com
##   5234.9899    584.0671   1095.4321  -6063.4638  -9655.5193  -2417.9728
##         ind
##  -3999.5006
```

- The first model would be easier to interpret than the second one.

Transformation of Y

Log-transformation of Y

- Consider taking the logarithm of Y:

$$\log Y = \beta_0 + X\beta_1 + \epsilon$$

- In this regression model, the interpretation of β_1 is no longer the marginal effect of X on Y.

$$\beta_1 = \frac{\partial \log Y}{\partial X} = \frac{\partial \log Y}{\partial Y}\frac{\partial Y}{\partial X} \quad \text{(chain rule)}$$
$$= \frac{\partial Y / Y}{\partial X}$$

- Thus, $100 \times \beta_1$ is equal to the percent change in Y ($100 \times \partial Y / Y$ %) due to a one unit increase in X.

Transformation of Y

```
>   data$logprice <- log(data$price)
>   lm(logprice ~ area + floor + renov + stdist + com + ind, data)$coef %>%
+     round(4)
```

```
## (Intercept)        area       floor       renov      stdist         com
##      2.4381      0.0177      0.0179     -0.1800     -0.1212      0.0070
##         ind
##     -0.0410
```

- 1 m^2 increase in area size increases the property price by about 1.8%.
- 1 km increase in distance to railway station decreases the property price by about 12%.

Transformation of X

Change the scale/unit of X

- For example, consider redefining `stdist` by measuring it in meters, rather than in kilometers. The new variable is $X^* = 1000X$.
- When an explanatory variable is multiplied by a constant c, then the corresponding coefficient's estimate will be divided by c:

$$Y = \beta_0 + X\beta_1 + \epsilon$$
$$Y = \beta_0 + X^*\beta_1/c + \epsilon \text{ where } X^* = cX$$

Transformation of X

```r
>   lm(price ~ area + floor + renov + stdist + com + ind, data)$coef %>%
+     round(4)
```

```
## (Intercept)        area       floor       renov      stdist         com
##      5.2350      0.5841      1.0954     -6.0635     -9.6555     -2.4180
##         ind
##     -3.9995
```

```r
>   data$stdist1 <- data$stdist*1000 # distance in meters
>   lm(price ~ area + floor + renov + stdist1 + com + ind, data)$coef %>%
+     round(4)
```

```
## (Intercept)        area       floor       renov     stdist1         com
##      5.2350      0.5841      1.0954     -6.0635     -0.0097     -2.4180
##         ind
##     -3.9995
```

- The first model would be easier to interpret than the second one.

Transformation of X

Log-transformation of X

- Take the logarithm of X:

$$Y = \beta_0 + (\log X)\beta_1 + \epsilon$$

- Again, in this regression model, β_1 cannot be interpreted as the marginal effect of X on Y.

$$\frac{\partial Y}{\partial X} = \frac{\partial \log X}{\partial X}\beta_1 \Longrightarrow \beta_1 = \frac{\partial Y}{\partial X / X}$$

- Thus, $\frac{1}{100} \times \beta_1$ is equal to how much Y changes due to a one percent increase in X $(100 \times \partial X / X\ \%)$.

Transformation of X

```
>   data$logarea <- log(data$area)
>   data$logstdist <- log(data$stdist)
>   lm(price ~ logarea + floor + renov + logstdist + com + ind, data)$coef
+     round(4)
```

```
## (Intercept)     logarea       floor       renov   logstdist         com
##    -55.6241     21.5885      1.2617     -5.9306     -2.9845     -3.3086
##         ind
##     -4.1793
```

- 1% increase in area size increases the property price by about 0.22 million JPY.
- 1% increase in distance to railway station decreases the property price by about 0.03 million JPY.

Log-log model

- Log-log model:

$$\log Y = \beta_0 + (\log X)\beta_1 + \epsilon$$

- In this model, β_1 represents the elasticity of Y with respect to X:

$\beta_1 =$ percent change in Y due to a one percent increase in X

$$\frac{\partial \log Y}{\partial X} = \frac{\partial \log X}{\partial X}\beta_1 \Longrightarrow \frac{\partial \log Y}{\partial Y}\frac{\partial Y}{\partial X} = \frac{\partial \log X}{\partial X}\beta_1$$
$$\Longrightarrow \beta_1 = \frac{\partial Y / Y}{\partial X / X}$$

- The log-log model is often used in economics to estimate a production function.
  - E.g., Cobb-Douglas production function: $Y = AL^{\beta}K^{\alpha}$. Taking the log of both sides leads to a log-log regression model.

Log-log model

```
>   lm(logprice ~ logarea + floor + renov + logstdist + com + ind, data)$co
+     round(4)
```

```
## (Intercept)     logarea       floor       renov   logstdist         com
##      0.4257      0.7427      0.0206     -0.1868     -0.0296     -0.0061
##         ind
##     -0.0516
```

- 1% increase in area size increases the property price by about 0.74%.
- 1% increase in distance to railway station decreases the property price by about 0.03%.

Summary

Model	Equation	Interpretation of β_1
Level-level model	$Y = \beta_0 + X\beta_1 + \epsilon$	1 unit increase in X increases Y by β_1
Log-level model	$\log Y = \beta_0 + X\beta_1 + \epsilon$	1 unit increase in X increases Y by $100\beta_1\%$
Level-log model	$Y = \beta_0 + \log(X)\beta_1 + \epsilon$	1 % increase in X increases Y by $\beta_1/100$
Log-log model	$\log Y = \beta_0 + \log(X)\beta_1 + \epsilon$	1 % increase in X increases Y by $\beta_1\%$

Which regression model(s) should be reported?

- As we have seen above, depending on the forms of variables, the interpretations of the estimation results may significantly differ.
- One ideal approach is to develop models based on some (economic) theory (e.g., Cobb-Douglas prod function).
- When no such theories are available, one should try multiple alternative models to see any differences between them and draw comprehensive conclusions.
    - Reporting only a single model is usually not recommended.
- Simply including all possible explanatory variables in the model improves the R-squared value, but the model's prediction performance may deteriorate (overfitting problem).

modelsummary

- When you would like to summarize several multiple regression results in a single table, the **modelsummary** package is very useful.

```
> library(modelsummary)
> models <- list()
> models[["level-level"]] <- lm(price ~ area + stdist + floor + renov, data)
> models[["log-level"]] <- lm(logprice ~ area + stdist + floor + renov, data)
> models[["log-log"]] <- lm(logprice ~ logarea + logstdist + floor + renov, data)
> modelsummary(models,
+              gof_omit = "Log.Lik.|AIC|BIC|F|RMSE",
+              stars = TRUE,
+              notes = "Standard errors in parentheses.",
+              output = "table.docx")
```

- The above code produces a word file “table.docx” that contains the following summary table of the regression results (next page).

modelsummary

	level-level	log-level	log-log
(Intercept)	3.598*	2.441***	0.417**
	(1.555)	(0.050)	(0.133)
area	0.588***	0.018***	
	(0.027)	(0.0009)	
stdist	-9.491***	-0.133+	
	(2.209)	(0.071)	
floor	1.055***	0.018***	0.020***
	(0.084)	(0.003)	(0.003)
renov	-6.070***	-0.180***	-0.187***
	(1.477)	(0.048)	(0.047)
logarea			0.741***
			(0.036)
logstdist			-0.035
			(0.030)
Num.Obs.	500	500	500
R2	0.688	0.591	0.598
R2 Adj.	0.686	0.587	0.595

Standard errors in parentheses.

+ $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$