# OLS Asymptotics| R programming language

## OLS Asymptotics

*In Ordinary Least Squares (OLS) estimation, asymptotics refer to the behavior of estimators as the sample size becomes large. Under the classical assumptions of the classical linear regression model (e.g., linearity, no perfect multicollinearity, homoscedasticity, and no autocorrelation), OLS estimators exhibit desirable asymptotic properties as the sample size approaches infinity:*

*Consistency: As the sample size increases, the OLS estimator converges in probability to the true value of the coefficient. That is, the estimator becomes unbiased and gives the correct value in large samples.*

*Mathematically, as $n \to \infty$ , the estimator $\hat{\beta}$ converges to $\beta$, the true parameter value.*

*Asymptotic Normality: The distribution of the OLS estimator becomes normal as the sample size grows. For large n, the OLS estimator follows a normal distribution with mean equal to the true parameter and variance equal to the asymptotic variance. This is important because it allows for inference using standard statistical tests. The distribution can be written as:*

$$\hat{\beta} \sim N(\beta, Var(\hat{\beta}))$$

*where the variance of $\hat{\beta}$ can be consistently estimated.*

*Efficiency: In large samples, OLS estimators are efficient in the class of linear unbiased estimators, meaning that they have the smallest possible variance among all linear unbiased estimators (this is a consequence of the Gauss-Markov theorem).*

*Asymptotics imply that, for large samples, OLS estimators are not only unbiased and consistent, but also normally distributed, making them suitable for hypothesis testing and constructing confidence intervals. However, these properties rely on the assumptions of the classical linear regression model being satisfied.*

*As the sample size increases, the standard errors decrease at the rate of $\sqrt{1/n}$, which is a key asymptotic property of OLS estimators. This is consistent with the fact that larger sample sizes lead to more efficient (precise) estimates of the coefficients.*

Script Editor

```
wage1 <- read.csv(paste0(directory, "wage1.csv"))

# Regression with full sample
model <- lm(wage ~ educ + tenure + exper, wage1)
summary(model)

(se1 <- vcov(model) %>% diag %>% sqrt %>% .["exper"])
(n1 <- nobs(model))

# Regression with half the sample
model_half <- lm(wage ~ educ + tenure + exper,
                 slice(wage1, 1:(n1/2-1)))
summary(model_half)

(se2 <- vcov(model_half) %>% diag %>% sqrt %>% .["exper"])
(n2 <- nobs(model_half))

se1/se2
sqrt(n2/n1)
```

# OLS Asymptotics| R programming language

Console

```
> # Regression with full sample
> model <- lm(wage ~ educ + tenure + exper, wage1)
> summary(model)

Call:
lm(formula = wage ~ educ + tenure + exper, data = wage1)

Residuals:
    Min      1Q  Median      3Q     Max
-7.6068 -1.7747 -0.6279  1.1969 14.6536

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.87273    0.72896  -3.941 9.22e-05 ***
educ         0.59897    0.05128  11.679  < 2e-16 ***
tenure       0.16927    0.02164   7.820 2.93e-14 ***
exper        0.02234    0.01206   1.853   0.0645 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.084 on 522 degrees of freedom
Multiple R-squared:  0.3064,    Adjusted R-squared:  0.3024
F-statistic: 76.87 on 3 and 522 DF,  p-value: < 2.2e-16

> (se1 <- vcov(model) %>% diag %>% sqrt %>% .["exper"])
     exper
0.01205685
> (n1 <- nobs(model))
[1] 526
> # Regression with half the sample
> model_half <- lm(wage ~ educ + tenure + exper,
+                  slice(wage1, 1:(n1/2-1)))
> summary(model_half)

Call:
lm(formula = wage ~ educ + tenure + exper, data = slice(wage1,
    1:(n1/2 - 1)))

Residuals:
    Min      1Q  Median      3Q     Max
-9.1233 -1.9799 -0.4104  1.1393 13.7413

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -4.81508    1.23701  -3.893 0.000126 ***
educ         0.73164    0.08753   8.359 3.98e-15 ***
tenure       0.20928    0.03586   5.837 1.59e-08 ***
exper        0.04766    0.01942   2.454 0.014777 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.376 on 258 degrees of freedom
Multiple R-squared:  0.3294,    Adjusted R-squared:  0.3216
F-statistic: 42.24 on 3 and 258 DF,  p-value: < 2.2e-16

> (se2 <- vcov(model_half) %>% diag %>% sqrt %>% .["exper"])
     exper
0.01942017
> (n2 <- nobs(model_half))
[1] 262
> se1/se2
    exper
0.6208416
> sqrt(n2/n1)
[1] 0.7057612
```

*The two ratios are quite similar (0.6208 vs. 0.7058), which aligns with the asymptotic theory that standard errors decrease at the rate of √(1/n)*