Lecture 3: Designing simulations

Last time

Potential results: Coverage: $\varepsilon_i \sim$ Normal 95% $\varepsilon_i \sim$ Exp 95% $\varepsilon_i \sim \chi^2$ 95%

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

How would you study the importance of the normality assumption?

One approach:

- Simulate data with different distributions for ε_i
  e.g. Normal, χ^2, exponential, etc.
- Fit the linear regression model and calculate a 95% confidence interval for β_1. Ideally, 95% of these intervals contain β_1
- Repeat many times; does the confidence interval have the desired coverage (e.g. do 95% of the intervals actually contain β_1)?
- Compare coverage for different distribution for ε_i

If coverage is << 95%, or >> 95%, the normality assumption is important!

Simulating data

To start, simulate data for which the normality assumption holds:

```r
n <- 100 # sample size
beta0 <- 0.5 # intercept
beta1 <- 1 # slope

x <- runif(n, min=0, max=1)
noise <- rnorm(n, mean=0, sd=1)
y <- beta0 + beta1*x + noise
```

$(X_1, Y_1), \ldots, (X_n, Y_n)$ (n observations)

$\beta_0 = 0.5$

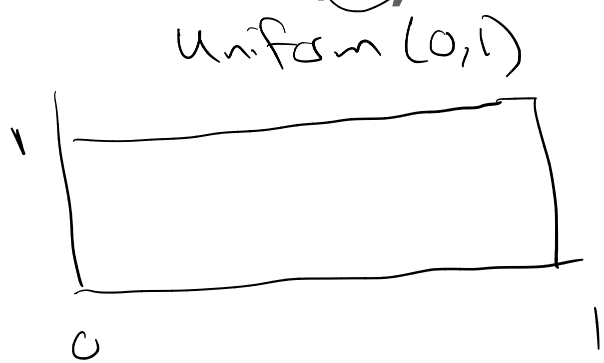
$\beta_1 = 1$

$X_i \sim \text{Uniform}(0,1)$

$\varepsilon_i \sim \text{Normal}(0,1)$

$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$

- `runif(n, min=0, max=1)` samples X_i uniformly between 0 and 1
- `rnorm(n, mean=0, sd=1)` samples $\varepsilon_i \sim N(0,1)$



Fit a model

```r
n <- 100 # sample size
beta0 <- 0.5 # intercept
beta1 <- 1 # slope

x <- runif(n, min=0, max=1)
noise <- rnorm(n, mean=0, sd=1)
y <- beta0 + beta1*x + noise

lm_mod <- lm(y ~ x)
lm_mod
```

generating data (xs and ys)

response

explanatory

```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
     0.2836       1.4302
```

$\hat{\beta}_0$ $\hat{\beta}_1$

95% CI for β_1:

$$\hat{\beta}_1 \pm t^*_{n-2} SE_{\hat{\beta}_1}$$

≈ 2

Calculate confidence interval

```
lm_mod <- lm(y ~ x)

ci <- confint(lm_mod, "x", level = 0.95)
ci
```

fitted model

95% CI

```
       2.5 %   97.5 %
x 0.6883911 2.172003
```

coefficient of interest (eg. β_1)

- **Question:** How can we check whether the confidence interval contains the true β_1 ?

$\beta_1 = 1$

0.688 < 1 & 2.172 > 1 (TRUE)

ci[1] < 1 & ci[2] > 1

Calculate confidence interval

```r
lm_mod <- lm(y ~ x)

ci <- confint(lm_mod, "x", level = 0.95)
ci
```

```
      2.5 %   97.5 %
x 0.6883911 2.172003
```

- **Question:** How can we check whether the confidence interval contains the true β_1 ?

```r
ci[1] < 1 & ci[2] > 1
```

```
[1] TRUE
```

ci[1] < beta1 & ci[2] > beta1

Repeat!

```
nsim <- 1000
n <- 100 # sample size
beta0 <- 0.5 # intercept
beta1 <- 1 # slope
results <- rep(NA, nsim)

for(i in 1:nsim){
  x <- runif(n, min=0, max=1)
  noise <- rnorm(n, mean=0, sd=1)
  y <- beta0 + beta1*x + noise

  lm_mod <- lm(y ~ x)
  ci <- confint(lm_mod, "x", level = 0.95)

  results[i] <- ci[1] < 1 & ci[2] > 1
}
mean(results)
```

sample data at each iteration

fit model, calculate a 95% CI

check if CI contains β_1, store result

- What fraction of the time should the confidence interval contain β_1 ?

expect ≈ 0.95

Repeat!

```
nsim <- 1000
n <- 100 # sample size
beta0 <- 0.5 # intercept
beta1 <- 1 # slope
results <- rep(NA, nsim)

for(i in 1:nsim){
  x <- runif(n, min=0, max=1)
  noise <- rnorm(n, mean=0, sd=1)
  y <- beta0 + beta1*x + noise

  lm_mod <- lm(y ~ x)
  ci <- confint(lm_mod, "x", level = 0.95)

  results[i] <- ci[1] < 1 & ci[2] > 1
}
mean(results)
```

```
[1] 0.952
```

next step: try a different distribution for ε_i

- What should we do next?

Class activity

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

That is, how important is the assumption that $\varepsilon_i \sim N(0, \sigma^2)$?

Continue simulation from last time, but experiment with different values of n and different distributions for the noise term.

https://sta279-f23.github.io/class_activities/ca_lecture_3.html

Class activity

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

How does confidence interval coverage change when you change the distribution of ε_i?

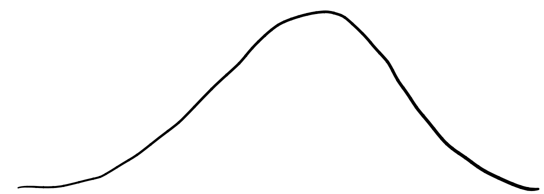
Class activity

```
nsim <- 1000
n <- 100 # sample size
beta0 <- 0.5 # intercept
beta1 <- 1 # slope
results <- rep(NA, nsim)

for(i in 1:nsim){
  x <- runif(n, min=0, max=1)
  noise <- rchisq(n, 1)
  y <- beta0 + beta1*x + noise

  lm_mod <- lm(y ~ x)
  ci <- confint(lm_mod, "x", level = 0.95)

  results[i] <- ci[1] < 1 & ci[2] > 1
}
mean(results)
```



χ^2_1

```
[1] 0.963
```

Normal = 95%

Exp(1) $\approx$ 95%

χ^2_1 $\approx$ 95%

ADEMP: A useful framework for simulation studies

- **Aims:** Why are we doing the study?
- **Data generation:** How are the data simulated?
- **Estimand/target:** What are we estimating for each simulated dataset?
- **Methods:** What methods are we using for model fitting, estimation, etc?
- **Performance measures:** How do we measure performance of our chosen methods?

ADEMP

For the normal errors simulation study:

- **Aims:** Assess importance of the normality assumption
- **Data generation:** $X_i \sim Uniform(0,1)$ $Y_i = 0.5 + X_i + \varepsilon_i$
  $\varepsilon_i \sim Uniform(0,1)$ or $\varepsilon_i \sim Exp(1)$
  or $\varepsilon_i \sim N(0,1)$ or $\varepsilon_i \sim \chi^2_1$ ...
- **Estimand/target:** β_1
- **Methods:** Fit linear model in R, calculate 95% CIs for β_1
- **Performance measures:** observed coverage of confidence intervals