Robust Standard Errors - Solutions

Exercise

1. Replicate the simulation from the right panel of my plot above:
   a. Set the seed to `4321` and generate `n = 100` uniform draws for `x`
   b. Set `y` equal to `0.2 + 0.9 * x + error` where `error` is a vector of independent, mean-zero normal errors with standard deviation `sqrt(2 * x)`.
   c. Replicate my plot and check that yours matches it.
2. Using `x` and `y`, replicate my regression and F-test results from above.
3. Use the formulas from earlier in this lecture to compute the "classical" and "HC0" standard errors for the regression slope "by hand" based on `x` and `y`. Check that your results match those of `lm_robust()`.

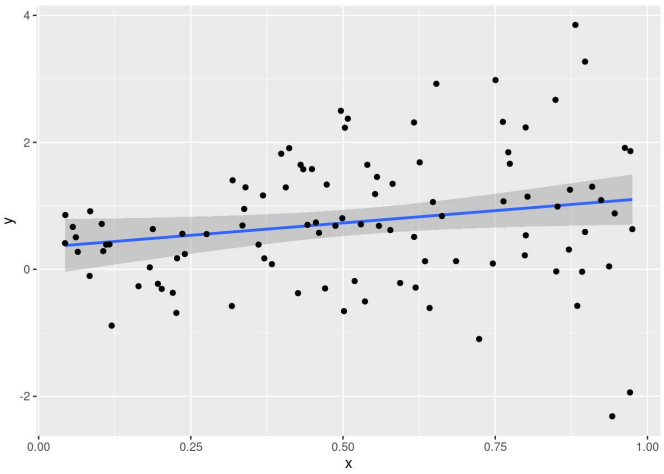
Solution

Part 1

```
library(tidyverse)
set.seed(4321)

n <- 100
x <- runif(n)
error <- rnorm(n, mean = 0, sd = sqrt(2 * x))
intercept <- 0.2
slope <- 0.9
y <- intercept + slope * x + error

tibble(x, y) |>
  ggplot(aes(x, y)) +
  geom_smooth(method = 'lm') +
  geom_point()
```



Part 2

```
library(estimatr)
library(car)
library(broom)
library(modelsummary)

reg_classical <- lm_robust(y ~ x, se_type = 'classical')

reg_robust <- lm_robust(y ~ x, se_type = 'HC0')

modelsummary(list(Classical = reg_classical, Robust = reg_robust),
             fmt = 2,
             gof_omit = 'R2 Adj.|AIC|BIC')
```

	Classical	Robust
(Intercept)	0.34	0.34
	(0.22)	(0.17)
x	0.78	0.78
	(0.38)	(0.40)
Num.Obs.	100	100
R2	0.041	0.041

	Classical	Robust
RMSE	1.03	1.03

```
linearHypothesis(reg_classical, 'x = 0') |> tidy()
```

```
# A tibble: 1 × 8
  term  null.value estimate std.error statistic p.value df.residual    df
  <chr>      <dbl>    <dbl>     <dbl>     <dbl>   <dbl>       <dbl> <dbl>
1 x              0    0.777     0.379      4.21  0.0402          98     1
```

```
linearHypothesis(reg_robust, 'x = 0') |> tidy()
```

```
# A tibble: 1 × 8
  term  null.value estimate std.error statistic p.value df.residual    df
  <chr>      <dbl>    <dbl>     <dbl>     <dbl>   <dbl>       <dbl> <dbl>
1 x              0    0.777     0.403      3.72  0.0538          98     1
```

Part 3

```
reg <- lm(y ~ x)
uhat <- residuals(reg)
x_demeaned <- x - mean(x)
n <- length(uhat)

# Classical
sigma_sq_hat <- sum(uhat^2) / (n - 2) # two estimated parameters
var_classical <- sigma_sq_hat / sum(x_demeaned^2)
SE_classical <- sqrt(var_classical)

c(lm_robust = tidy(reg_classical) |>
  filter(term == 'x') |>
  pull(std.error),
  by_hand = SE_classical)
```

```
lm_robust   by_hand
 0.378506  0.378506
```

```
# HC0
var_HC0 <- sum(uhat^2 * x_demeaned^2) / (sum(x_demeaned^2)^2)
SE_HC0 <- sqrt(var_HC0)

c(lm_robust = tidy(reg_robust) |>
  filter(term == 'x') |>
  pull(std.error),  by_hand = SE_HC0)
```

```
lm_robust   by_hand
0.4027311 0.4027311
```