Outline

Revisit the CEB dataset

- Number of Children Ever Born (CEB) to Women of Indian Race By Marital Duration, Type of Place of Residence and Educational Level
- Each cell shows the mean, variance, and sample size.

Marr.	Suva				Urban				Rural			
Dur.	N	LP	UP	S+	N	LP	UP	S+	N	LP	UP	S+
0–4	0.50	1.14	0.90	0.73	1.17	0.85	1.05	0.69	0.97	0.96	0.97	0.74
	1.14	0.73	0.67	0.48	1.06	1.59	0.73	0.54	0.88	0.81	0.80	0.59
	8	21	42	51	12	27	39	51	62	102	107	47
5–9	3.10	2.67	2.04	1.73	4.54	2.65	2.68	2.29	2.44	2.71	2.47	2.24
	1.66	0.99	1.87	0.68	3.44	1.51	0.97	0.81	1.93	1.36	1.30	1.19
	10	30	24	22	13	37	44	21	70	117	81	21
10–14	4.08	3.67	2.90	2.00	4.17	3.33	3.62	3.33	4.14	4.14	3.94	3.33
	1.72	2.31	1.57	1.82	2.97	2.99	1.96	1.52	3.52	3.31	3.28	2.50
	12	27	20	12	18	43	29	15	88	132	50	9
15–19	4.21	4.94	3.15	2.75	4.70	5.36	4.60	3.80	5.06	5.59	4.50	2.00
	2.03	1.46	0.81	0.92	7.40	2.97	3.83	0.70	4.91	3.23	3.29	–
	14	31	13	4	23	42	20	5	114	86	30	1
20–24	5.62	5.06	3.92	2.60	5.36	5.88	5.00	5.33	6.46	6.34	5.74	2.50
	4.15	4.64	4.08	4.30	7.19	4.44	4.33	0.33	8.20	5.72	5.20	0.50
	21	18	12	5	22	25	13	3	117	68	23	2
25–29	6.60	6.74	5.38	2.00	6.52	7.51	7.54	–	7.48	7.81	5.80	–
	12.40	11.66	4.27	–	11.45	10.53	12.60	–	11.34	7.57	7.07	–
	47	27	8	1	46	45	13	–	195	59	10	–

Additive model

TABLE 4.4: Estimates for Additive Log-Linear Model of Children Ever Born by Marital Duration, Type of Place of Residence and Educational Level

Parameter		Estimate	Std. Error	z-ratio
Constant		−0.1173	0.0549	−2.14
Duration	0–4	–		
	5–9	0.9977	0.0528	18.91
	10–14	1.3705	0.0511	26.83
	15–19	1.6142	0.0512	31.52
	20–24	1.7855	0.0512	34.86
	25–29	1.9768	0.0500	39.50
Residence	Suva	–		
	Urban	0.1123	0.0325	3.46
	Rural	0.1512	0.0283	5.34
Education	None	–		
	Lower	0.0231	0.0227	1.02
	Upper	-0.1017	0.0310	-3.28
	Sec+	-0.3096	0.0552	-5.61

ANOVA model

TABLE 4.3: Deviances for Poisson Log-linear Models Fitted to the Data on CEB by Marriage Duration, Residence and Education

Model	Deviance	d.f.
Null	3731.52	69
One-factor Models		
Duration	165.84	64
Residence	3659.23	67
Education	2661.00	66
Two-factor Models		
$D + R$	120.68	62
$D + E$	100.01	61
DR	108.84	52
DE	84.46	46
Three-factor Models		
$D + R + E$	70.65	59
$D + RE$	59.89	53
$E + DR$	57.06	49
$R + DE$	54.91	44
$DR + RE$	44.27	43
$DE + RE$	44.60	38
$DR + DE$	42.72	34
$DR + DE + RE$	30.95	28

Interpretation

- Null model has a deviance of 3732 on 69 degrees of freedom, which does not pass the goodness-of-test. $\Rightarrow$ reject the hypothesis that "the expect number of children is the same for all these groups".
- Introducing marital duration leads to substantial reduction of 3566 at only 5 d.f. $\Rightarrow$ significant effect of "duration" on the number of children
- The additive model D+R+E has a deviance of 70.65 on 59 d.f. The associated P-value under a χ^2 distribution is 0.14, so the model provides a good description of the data.
- Education effect:
  - compare model *E* to model Null (1071 on 3 d.f.)
  - compare model D+E to model D (65.8 on 3 d.f.)
  - compare model D+R+E to model D+R. (50.1 on 3 d.f.)
  - part of education effect may be attributed to the fact that more educated women tend to live in Suva or in other urban areas (collinearity between E and R).

Interaction effect

- Does education make more of a difference in rural areas than in urban areas?
- Compare $D + R + E$ to $D + RE$. $\Rightarrow$ reduces the deviance by 10.8 at the expense of 6 d.f. $\Rightarrow$ not significant, with a P-value of 0.096.
- Does education effect increase with marital duration?
- Compare $D + R + E$ to $R + DE$ $\Rightarrow$ reduces the deviance by 15.7 at the expense of 15 d.f. $\Rightarrow$ hardly a bargain.

Outline

Interaction between 2 categorical predictors

Two-way interaction between X_1 (k_1 levels, think area) and X_2 (k_2 levels, think education): when the mean response (Y, or log odds, or log intensity) differences between X_1 levels depend on the level of X_2.

- With *no* interaction, there are constraints on group means. X_1 requires $k_1 - 1$ extra coefficients (other than intercept), X_2 requires $k_2 - 1$ extra coefficients.
- In R, $X_1 * X_2$ is a shortcut for $X_1 + X_2 + X_1 : X_2$.
- The interaction $X_1 : X_2$ requires $(k_1 - 1) * (k_2 - 1)$ df, i.e. extra coefficients.
- *With* interaction, total # of coefficients (including intercept) = total # of groups, $k_1 * k_2$.

Degrees of freedom

For model $Y \sim X_1 * X_2$:

source	df
intercept	1
X_1	$k_1 - 1$
X_2	$k_2 - 1$
$X_1 : X_2$	$(k_1 - 1)(k_2 - 1)$
total # coefs	$k_1 * k_2$
residual	$n - k_1 k_2$

where n is the number of observations (rows) in the data.

Interactions between 3 categorical predictors

Two-way interactions between X_1, X_2 and X_3 (k_3 levels, think of marriage duration):

- $(X_1 : X_2)$ is the 2-way interaction between X_1 and X_2 when $X_3 = 0$ or reference level.
- $(X_1 : X_3)$ is the 2-way interaction between X_1 and X_3 when $X_2 = 0$ or reference level.
- $(X_2 : X_3)$ is the 2-way interaction between X_2 and X_3 when $X_1 = 0$ or reference level.

There is a three-way interaction $(X_1 : X_2 : X_3)$ if the interaction coefficient(s) $(X_1 : X_2)$ depend on the level of X_3, or, equivalently, if the $(X_2 : X_3)$ interaction coefficient(s) depend on the level of X_1, or if the $(X_1 : X_3)$ interaction coefficient(s) depends on the level of X_2.

- With the 3-way interaction and all 2-way interactions, the total # of coefficients (including intercept) equals the total # of groups, $k_1 * k_2 * k_3$.

Degrees of freedom

For $Y \sim X_1 * X_2 * X_3$:

source	df
intercept	1
X_1	$k_1 - 1$
X_2	$k_2 - 1$
$X_1 : X_2$	$(k_1 - 1)(k_2 - 1)$
$X_1 : X_3$	$(k_1 - 1)(k_3 - 1)$
$X_2 : X_3$	$(k_2 - 1)(k_3 - 1)$
$X_1 : X_2 : X_3$	$(k_1 - 1)(k_2 - 1)(k_3 - 1)$
total # coef	$k_1 * k_2 * k_3$
residual	$n - k_1 k_2 k_3$

where n is the number of observations (rows) in the data.

Notations in R

(and many other programs)

These are equivalent notations for model formulas:

- $X_1 * X_2$ and $X_1 + X_2 + X_1 : X_2$.
- Main effects and one 2-way interaction: $X_1 * X_2 + X_3$ and $X_1 + X_2 + X_1 : X_2 + X_3$
- Mains effects and two 2-way interactions: $X_1 * X_2 + X_3 + X_2 : X_3$ and $X_1 + X_2 + X_3 + X_1 : X_2 + X_2 : X_3$ and $X_1 + X_2 * X_3 + X_1 : X_2$
- Main effects and all 2-way interactions: $X_1 * X_2 + X_3 + X_2 : X_3 + X_1 : X_3$ and $X_1 + X_2 + X_3 + X_1 : X_2 + X_1 : X_3 + X_2 : X_3$
- All: $X_1 * X_2 * X_3$ and $X_1 + X_2 + X_3 + X_1 : X_2 + X_1 : X_3 + X_2 : X_3 + X_1 : X_2 : X_3$.

Hierarchy principle

Include all lower-level interactions.

- If we include $(X_1 : X_2)$, then we must also include X_1 and X_2.

  If we allow X_1 to have an effect that depends on the level of X_2, it makes little sense to assume that X_1 has no effect when $X_2 = 0$ or reference level.
- If we include $(X_1 : X_2 : X_3)$ then we must also include all three 2-way interactions $(X_1 : X_2,\ X_1 : X_3,\ X_2 : X_3)$ and all three main effects (X_1, X_2, X_3).
- A predictor X_1 is said to have an effect as soon as X_1 is involved in the main-effect term or any interaction-effect term

Outline

Why use simulations?

to assess uncertainty in predictions, in estimated regression coefficients, etc. This is in contrast to deriving formulas for standard errors.

pros:

- we do not need to learn how to derive formulas.
- we can use simulations even when formulas are only approximations
- we can assess uncertainty in quantities for which there are not formulas.

cons:

- we will learn how to write small programs in `R`.
- we will do parametric bootstrapping.

Runoff data

We investigate if the storm produces any measurable runoff. The predictors are total amount of precipition (inches) and maximum intensity at 10 minutes (in/min).

```
> runoff #  check the data set
    Precip MaxIntensity10 RunoffEvent
1     0.47            0.96           1
2     0.34            0.18           0
3     0.16            0.24           0
...
186   0.11            0.18           1
...
231   1.75            2.70           1
```

Runoff data

We use logistic regression:

$$\text{Runoff}_i \sim \text{Ber}(\mu_i) \quad \text{independently},$$
$$\text{logit}(\mu_i) = \beta_0 + \beta_1 \text{Precip}_i + \beta_2 \text{MAxintensity10}_i$$

```
> fit2 = glm(RunoffEvent~Precip+MaxIntensity10,
             family=binomial, data=runoff)
> summary(fit2)
...
Residual deviance: 116.11  on 228  degrees of freedom

> fit2$df.residual
[1] 228
> fit2$deviance
[1] 116.10591                # seems too low
> pchisq(116.11, df=228)  # P(X2 < 116.11)
[1] 5.765843e-11
> curve(dchisq(x,df=228), from=100,to=300, xlab="x2")
> points(116.11, 0, col="red", pch=16)
```

Goodness of Fit Statistics (Recall from Slide 19)

Deviation of Model

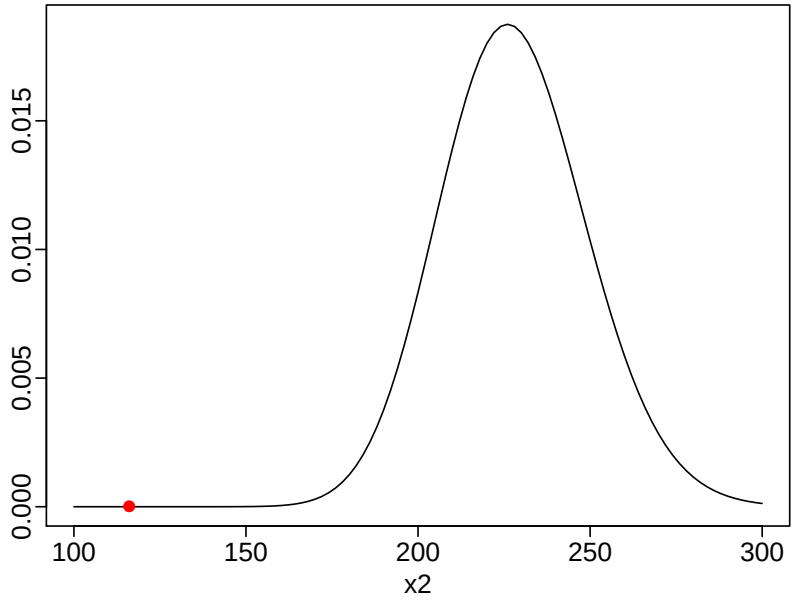
We define the deviation of a model:

$$\text{Dev}(\hat{\boldsymbol{\mu}}, \boldsymbol{Y}) \stackrel{\text{def}}{=} -2 \times \text{log-L}(\hat{\boldsymbol{\mu}}, \boldsymbol{Y}) - 2 \times \text{log-L}(\boldsymbol{Y}, \boldsymbol{Y})$$

where $\hat{\boldsymbol{\mu}}$ denotes the fitted mean based on the specified model and $\boldsymbol{Y}$ the observations.

- Deviation is used to assess the goodness of fit of the model.
- As $n \to \infty$, $\text{Dev}(\hat{\mu}, \boldsymbol{Y}) \sim \chi^2_{n-p}$ under the true generative model.

Is there real underdispersion?



Why is the deviance so low? Also, $\hat{\sigma}^2 = 0.65$. Underdispersion?

```
> sum(residuals(fit2, type="pearson")^2 )/228
[1] 0.6524366
```

Recall from slide 17

The over-dispersion parameter σ^2 can be estimated using Pearson's X^2:

$$\hat{\phi} = \frac{1}{n-p} \sum_{i=1}^{n} \frac{(Y_i - \hat{\mu}_i)^2}{V(\hat{\mu}_i)}.$$

Simulating the distribution of the residual deviance

"Formula": When sample sizes are large and if the model is correct, then the residual deviance is approximately χ^2 distributed, on residual df.

Here: deviance$=$ 116.11 on df$=$ 228. The "formula" gives us a very low p-value and suggests underdispersion. But can we trust the large-sample (i.e., asymptotical) distribution formula ?

Parametric Bootstrapping

To find the "typical" distribution of the statistic of interest (in this case, deviance), we can take a simulation-based procedure:

Parametric Bootstrapping

1. simulate new data sets under our hypothesized model: use our estimated model and repeat many experiments *in silico*. Our model will be correct (H_0 true) for each of these experiments.
2. apply the same analysis as we did for the original data set,
3. calculate the the statistics of interest for each of these new data sets,
4. repeat many times, summarize the statistics of interest.

Simulating one new data set

```
> dim(runoff)   # 231 11
> mu=predict(fit2, type="response") # estimated probabilities
> mu                              # of runoff events for each storm
    1     2     3     4     5     6     7     8     9    10    11
0.140 0.026 0.018 0.028 0.018 0.051 0.021 0.249 0.025 0.927 0.547
...
  222   223   224   225   226   227   228   229   230   231
0.675 0.638 0.099 0.031 0.763 0.689 0.899 1.000 0.257 0.993

> sim.events = rbinom(231, size=1, prob=mu)    # One experiment
> sim.events                                   # was simulated
  [1] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 ...
 [38] 1 0 1 0 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 ...
...
[223] 1 0 0 1 1 1 1 1 1

> sim.data = data.frame(
+    Precip         = runoff$Precip,
+    MaxIntensity10 = runoff$MaxIntensity10,
+    RunoffEvent    = sim.events
+ )
```

Simulating one new data set

```
> sim.data  # just to check the new data set
    Precip MaxIntensity10 RunoffEvent
1     0.47            0.96           0
2     0.34            0.18           0
3     0.16            0.24           0
...
186   0.11            0.18           1
187   0.16            0.48           0
...
230   0.58            1.20           1
231   1.75            2.70           1

# now apply the same analysis
> sim.fit = glm(RunoffEvent ~ Precip + MaxIntensity10,
+               data=sim.data, family=quasibinomial)
> sim.fit$deviance
[1] 141.73432  # this is random: from one experiment.
```

We got this 141.7 deviance "just by chance" under the H_0: the model (binomial, Precip + MaxIntensity10) was true.

Simulating many new data sets

First define a function to make and analyze a single data set:

```
> simulate.deviance = function(){
+  sim.data = data.frame(
+     Precip          = runoff$Precip,
+     MaxIntensity10 = runoff$MaxIntensity10,
+     RunoffEvent     = rbinom(231, size=1, prob=mu)
+  )
+  sim.fit = glm(RunoffEvent ~ Precip + MaxIntensity10,
+                data=sim.data, family=binomial)
+  return( sim.fit$deviance )
+ }

> simulate.deviance() # check that this function works
[1] 87.839458         # this is random
> simulate.deviance()
[1] 111.6603          # new random deviance from new expt
> simulate.deviance()
[1] 102.7747          # from another new experiment
```

Simulating many new data sets

We replicate this simulation many times (1000 usually enough).

`replicate(n, function)` : needs a number `n` of times and a `function` to be repeated.

```
> sim1000 = replicate(1000, simulate.deviance())
> sim1000
   [1] 109.1 102.3  94.1 108.8  73.3  94.9 115.1  99.2 114.9 118.9 ...
  [13] 102.9 102.5 105.9 135.1 120.0 150.1 131.5  93.7 100.3 118.3 ...
  [25] 141.9 109.7 103.8 128.4 113.0 148.5  92.5  72.4 106.6 114.9 ...
...
 [961] 114.4 117.6 108.0  91.6 103.5 129.5 110.0  96.8 100.6  92.3 ...
 [973]  99.3 116.7 109.9 109.5 112.5 107.9 133.3  94.5 117.4 134.9 ...
 [985] 111.6 107.6 108.7 135.0 114.4 127.7 100.8 106.2 112.5 129.1 ...
 [997] 109.1 122.4 105.6 143.3
```

Summarizing simulated deviances

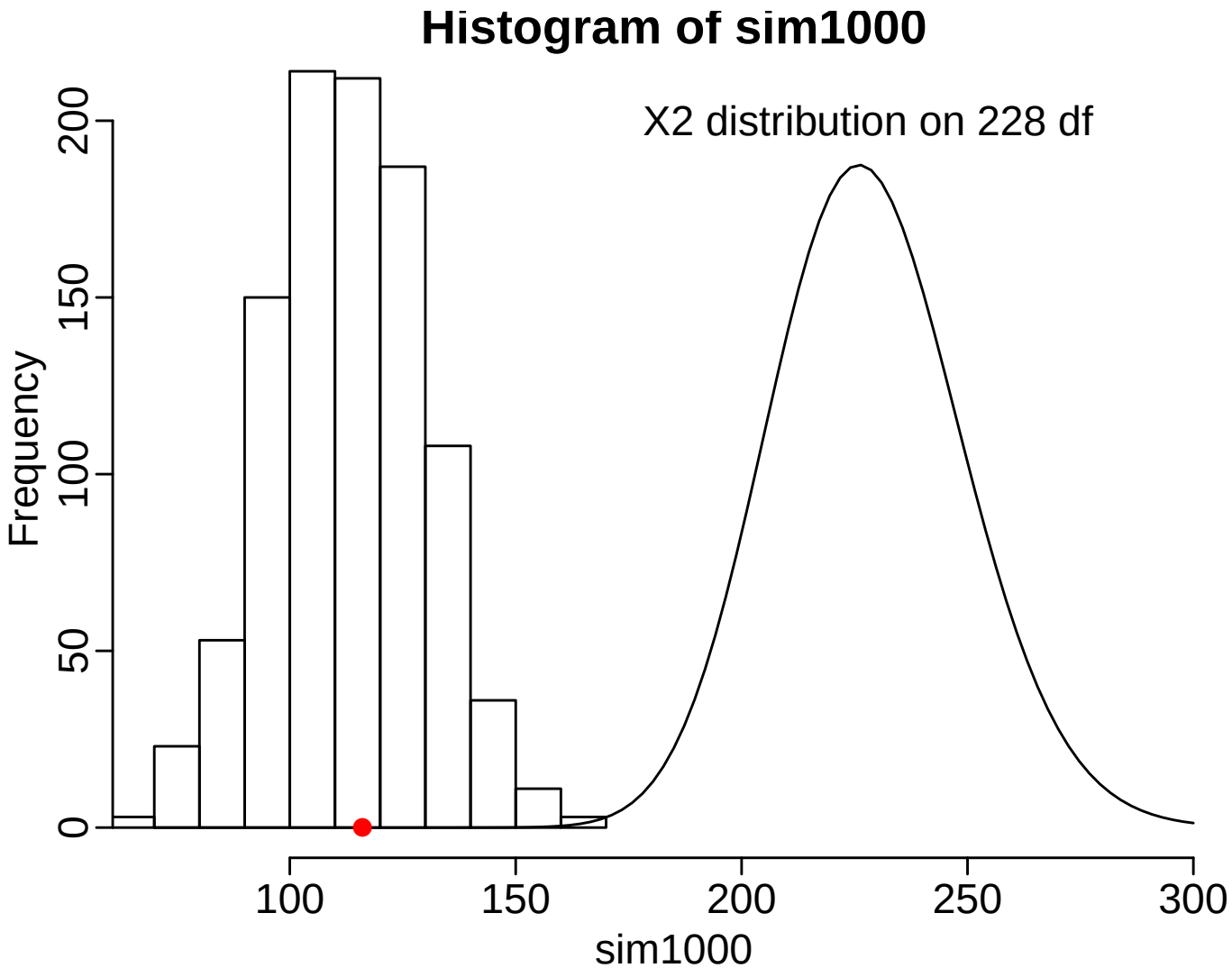
Now we summarize the 1000 deviance values, which were obtained "by chance" under our hypothesized model:

```
> hist(sim100)
```

Overlay the the simulated distribution with the theoretical χ^2 distribution... which we know is a bad approximation because the sample sizes are small.

```
> hist(sim1000, xlim=c(70,300))
> curve(1000*10*dchisq(x,df=228), add=T)
  # I used 1000 * 10 * chi-square_density   in order to match
  # the area under the curve (1000 * 10 * 1) with
  # the area of the histogram: 1000 points * width 10 of each bin.
> text("X2 distribution on 228 df", x=228, y=200)
> points(116.11, 0, col="red", pch=16)
```

Summarizing simulated deviances



No sign of lack of fit: so far, it looks like our binomial model is adequate.

Testing lack of fit with simulations

Conclusions:

- In the runoff experiment, the null distribution of the deviance under H_0: "our model is correct" is very far from a χ^2 distribution on df=residual df.
- In this case we should not trust the p-value obtained from comparing the residual deviance to the χ^2 distribution on residual df (was 5.10^{-11}).
- Instead, we should test H_0: "the model is correct" versus lack of fit with simulations.
- How often is the deviance even lower than 116.11 'just by chance'? We obtain p-value $= 0.57$: No lack of fit, No evidence of underdispersion.

Testing lack of fit with simulations

```
> sim1000 < 116.11
   [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
  [13]  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
...
 [985]  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
 [997]  TRUE FALSE  TRUE FALSE

> sum( (sim1000 < 116.11))
 # true=1 false=0, so the 'sum' of the true/false values
 # will be the number of 'true's.
[1] 573
> sum( (sim1000 < 116.11)) / 1000
[1] 0.573
```

p-value $= 0.57 =$ probability of observing a deviance of 116.11 or smaller, when the model is really true. Obtained by parametric boostrapping.

Simulation-based procedure vs. asymptotical formula

- We can use simulation to assess uncertainty in predictions or estimations.
- This is in contrast to deriving formulas for standard errors.
- One advantage is that we can assess uncertainty in quantities for which there are not formulas. E.g.
  - finding CI for ratio of estimated regression coefficients, e.g. $\frac{\hat{\beta}_0}{\hat{\beta}_1}$.
  - finding S.E. for $f(\hat{\beta}_0, \ldots, \hat{\beta}_p)$, where f is an arbitrary function of regression coefficients.
- A second advantage is we do not need to learn how to derive formulas.