# MA678 Homework 5

Jing Xu

10/25/2022

## 15.1 Poisson and negative binomial regression

The folder `RiskyBehavior` contains data from a randomized trial targeting couples at high risk of HIV infection. The intervention provided counseling sessions regarding practices that could reduce their likelihood of contracting HIV. Couples were randomized either to a control group, a group in which just the woman participated, or a group in which both members of the couple participated. One of the outcomes examined after three months was "number of unprotected sex acts."

### a)

Model this outcome as a function of treatment assignment using a Poisson regression. Does the model fit well? Is there evidence of overdispersion?

```
library(rosdata)
```

```
##
## Attaching package: 'rosdata'

## The following object is masked from 'package:MASS':
##
##     newcomb

## The following objects are masked from 'package:rstanarm':
##
##     kidiq, roaches, wells
```

```
data(risky)
risky$fupacts <- round(risky$fupacts)
risky$women_alone <- as.factor(risky$women_alone)
risky$couples <- as.factor(risky$couples)
fit_1a <- glm(fupacts ~ couples +women_alone, family = poisson(link = "log"), data = risky)
summary(fit_1a)
```

```
##
## Call:
## glm(formula = fupacts ~ couples + women_alone, family = poisson(link = "log"),
##     data = risky)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -6.6285  -4.9794  -3.2015   0.9847  27.1502
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   3.08960    0.01901  162.55   <2e-16 ***
## couples1     -0.32243    0.02737  -11.78   <2e-16 ***
## women_alone1 -0.57212    0.03023  -18.93   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 13299  on 433  degrees of freedom
## Residual deviance: 12925  on 431  degrees of freedom
## AIC: 14256
##
## Number of Fisher Scoring iterations: 6
```

```
#check for overdispersion
fit_1a$deviance/fit_1a$df.residual<=1
```

```
## [1] FALSE
```

```
#check for fitting
pchisq(fit_1a$deviance, fit_1a$df.residual, lower.tail = F)
```

```
## [1] 0
```

**b)**

Next extend the model to include pre-treatment measures of the outcome and the additional pre-treatment variables included in the dataset. Does the model fit well? Is there evidence of overdispersion?

```
fit_1b <- glm(fupacts ~ log(bupacts +1) + sex + couples + women_alone + bs_hiv, family = poisson(link =
summary(fit_1b)
```

```
##
## Call:
## glm(formula = fupacts ~ log(bupacts + 1) + sex + couples + women_alone +
##     bs_hiv, family = poisson(link = "log"), data = risky)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -11.425   -3.565   -1.898    1.003   20.832
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.02153    0.04792  21.317  < 2e-16 ***
## log(bupacts + 1)  0.66456    0.01217  54.596  < 2e-16 ***
## sexwoman          0.08181    0.02368   3.454 0.000551 ***
```

```
## couples1         -0.30894    0.02799 -11.038  < 2e-16 ***
## women_alone1     -0.50952    0.03031 -16.810  < 2e-16 ***
## bs_hivpositive   -0.40556    0.03543 -11.445  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 13298.6  on 433  degrees of freedom
## Residual deviance:  9184.3  on 428  degrees of freedom
## AIC: 10521
##
## Number of Fisher Scoring iterations: 6
```

```
#check for overdispersion
fit_1b$deviance/fit_1b$df.residual<=1
```

```
## [1] FALSE
```

```
#check for fitting
pchisq(fit_1b$deviance, fit_1b$df.residual, lower.tail = F)
```

```
## [1] 0
```

**c)**

Fit a negative binomial (overdispersed Poisson) model. What do you conclude regarding effectiveness of the intervention?

```
fit_1c <- glm.nb(fupacts ~ log(bupacts +1) + sex + couples + women_alone + bs_hiv, data = risky)
summary(fit_1c)
```

```
##
## Call:
## glm.nb(formula = fupacts ~ log(bupacts + 1) + sex + couples +
##     women_alone + bs_hiv, data = risky, init.theta = 0.4357586657,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0438  -1.3738  -0.4466   0.1795   2.6336
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.31804    0.23900   5.515 3.49e-08 ***
## log(bupacts + 1)  0.61832    0.06470   9.557  < 2e-16 ***
## sexwoman         -0.05974    0.14917  -0.400 0.688796
## couples1         -0.36679    0.18531  -1.979 0.047779 *
## women_alone1     -0.64007    0.18901  -3.386 0.000708 ***
## bs_hivpositive   -0.51314    0.18384  -2.791 0.005251 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.4358) family taken to be 1)
##
##     Null deviance: 603.09  on 433  degrees of freedom
## Residual deviance: 487.97  on 428  degrees of freedom
## AIC: 2953.3
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.4358
##           Std. Err.:  0.0330
##
##  2 x log-likelihood:  -2939.2650
```

```
#check for overdispersion
fit_1c$deviance/fit_1c$df.residual<=1
```

```
## [1] FALSE
```

```
#check for fitting
pchisq(fit_1c$deviance, fit_1c$df.residual, lower.tail = F)
```

```
## [1] 0.02364589
```

**d)**

These data include responses from both men and women from the participating couples. Does this give you any concern with regard to our modeling assumptions?

there's a linear relationship between these two variables, so the models don't fit well

## 15.3 Binomial regression

Redo the basketball shooting example on page 270, making some changes:

**(a)**

Instead of having each player shoot 20 times, let the number of shots per player vary, drawn from the uniform distribution between 10 and 30.

```
set.seed(100)
N <- 100
height <- rnorm(N, 72, 3)
p <- 0.4 + 0.1*(height - 72)/3
n <- floor(runif(N, min=10, max=31))
y <- rbinom(N, n, p)
data <- data.frame(n = n, y = y, height = height)
```

**(b)**

Instead of having the true probability of success be linear, have the true probability be a logistic function, set so that Pr(success) = 0.3 for a player who is 5'9" and 0.4 for a 6' tall player.

```
b1 <- (logit(.4) - logit(.3))/3
b0 <- (logit(.4) - 72*b1)
set.seed(100)
N <- 100
height <- rnorm(N, 72, 3)
p <- invlogit(b0 + b1*height)
n <- floor(runif(N, min=10, max=31))
y <- rbinom(N, n, p)
data <- data.frame(n = n, y = y, height = height)
```

## 15.7 Tobit model for mixed discrete/continuous data

Experimental data from the National Supported Work example are in the folder `Lalonde`. Use the treatment indicator and pre-treatment variables to predict post-treatment (1978) earnings using a Tobit model. Interpret the model coefficients.

```
lalonde <- haven::read_dta("http://www.nber.org/~rdehejia/data/nsw_dw.dta")
fit_7 <- censReg(formula = re78 ~ re75 + re74, data = lalonde)
summary(fit_7)
```

```
##
## Call:
## censReg(formula = re78 ~ re75 + re74, data = lalonde)
##
## Observations:
##          Total  Left-censored     Uncensored Right-censored
##            445            137            308              0
##
## Coefficients:
##              Estimate Std. error t value  Pr(> t)
## (Intercept) 3.092e+03  4.901e+02   6.309 2.81e-10 ***
## re75        1.731e-01  1.785e-01   0.970    0.332
## re74        7.553e-02  1.051e-01   0.719    0.472
## logSigma    9.076e+00  4.281e-02 211.997  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Newton-Raphson maximisation, 8 iterations
## Return code 8: successive function values within relative tolerance limit (reltol)
## Log-likelihood: -3348.565 on 4 Df
```

## 15.8 Robust linear regression using the t model

The folder `Congress` has the votes for the Democratic and Republican candidates in each U.S. congressional district in 1988, along with the parties' vote proportions in 1986 and an indicator for whether the incumbent was running for reelection in 1988. For your analysis, just use the elections that were contested by both parties in both years.

```
library(rosdata)
data(congress)
```

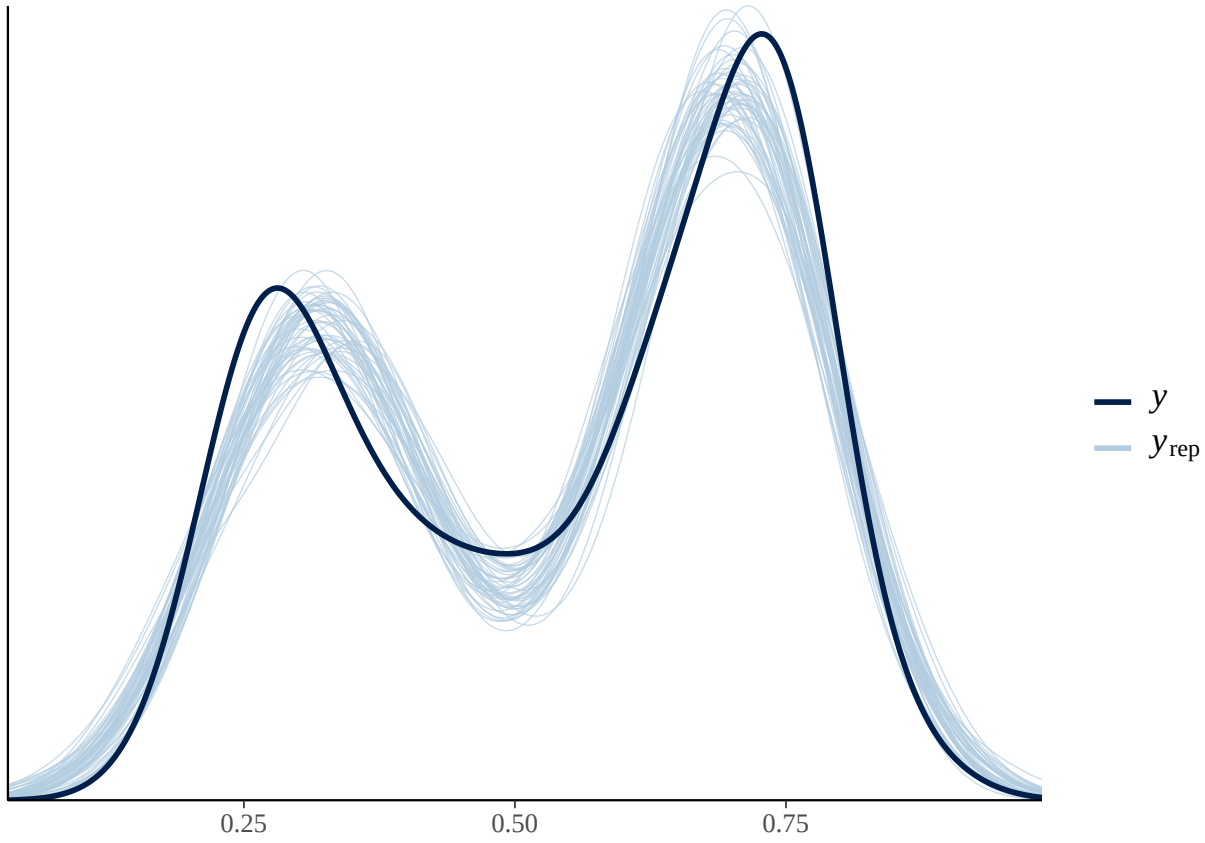
### (a)

Fit a linear regression using `stan_glm` with the usual normal-distribution model for the errors predicting 1988 Democratic vote share from the other variables and assess model fit.

```
df <- data.frame(vote = congress$v88_adj, past_vote = congress$v86_adj, inc = congress$inc88)
fit_8a <- stan_glm(vote ~ past_vote + inc, data = df, refresh = 0)
print(fit_8a)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      vote ~ past_vote + inc
##  observations: 435
##  predictors:   3
## ------
##             Median MAD_SD
## (Intercept) 0.2    0.0
## past_vote   0.5    0.0
## inc         0.1    0.0
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 0.1    0.0
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

```
pp_check(fit_8a)
```




**(b)**

Fit the same sort of model using the `brms` package with a $t$ distribution, using the `brm` function with the student family. Again assess model fit.

```
# fit_8b <- brm(vote ~ past_vote + inc, family = student(link = "identity"), data = df )
```

**(c)**

Which model do you prefer?

## 15.9 Robust regression for binary data using the robit model

Use the same data as the previous example with the goal instead of predicting for each district whether it was won by the Democratic or Republican candidate.

**(a)**

Fit a standard logistic or probit regression and assess model fit.

```
fit_9a <- stan_glm(v88_adj ~ v86_adj + inc86, family = binomial(link = "probit"), data = congress, refr
```

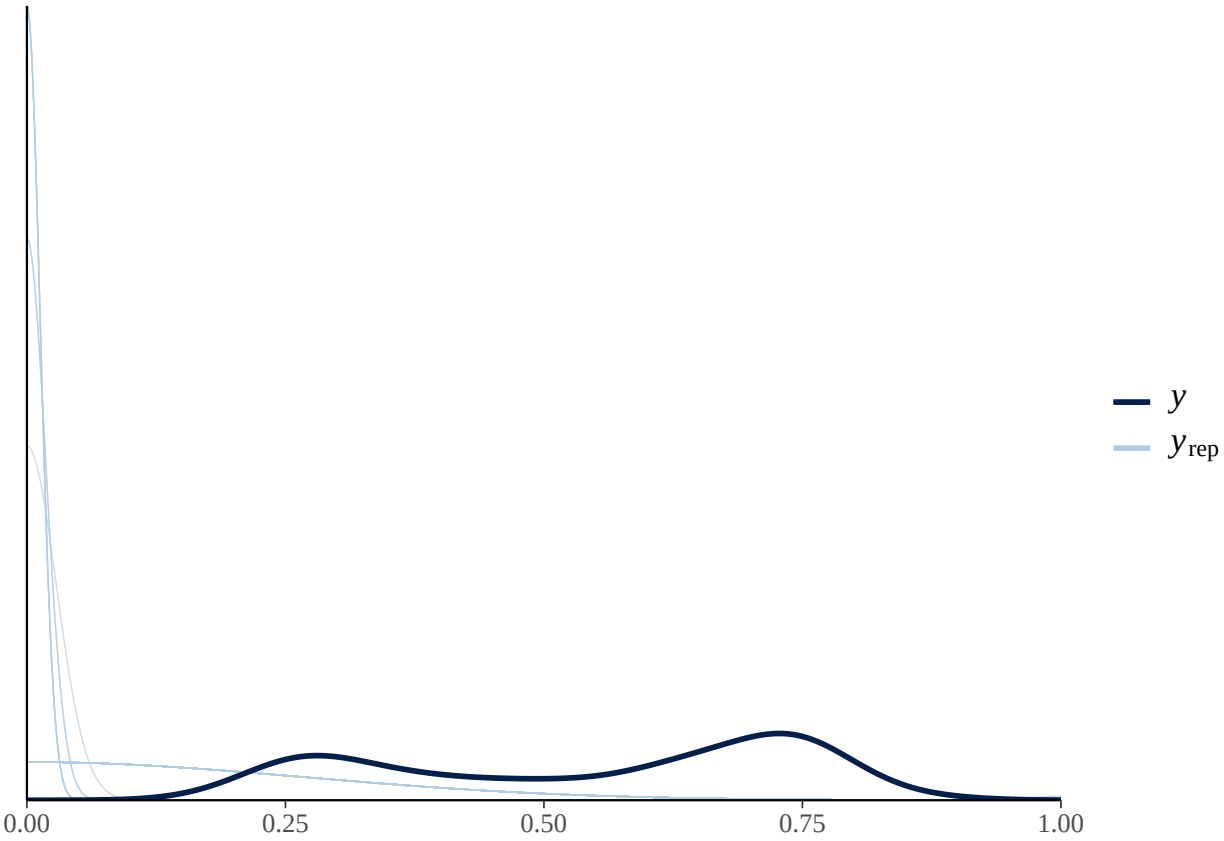
```
## Warning: There were 2 divergent transitions after warmup. See
## https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup
## to find out why this is a problem and how to eliminate them.
```

```
## Warning: Examine the pairs() plot to diagnose sampling problems
```

```
summary(fit_9a)
```

```
##
## Model Info:
##  function:     stan_glm
##  family:       binomial [probit]
##  formula:      v88_adj ~ v86_adj + inc86
##  algorithm:    sampling
##  sample:       4000 (posterior sample size)
##  priors:       see help('prior_summary')
##  observations: 435
##  predictors:   3
##
## Estimates:
##               mean   sd    10%   50%   90%
## (Intercept)  -7.5    6.1 -16.0  -6.8  -0.4
## v86_adj       0.7   10.2 -11.9   0.4  13.8
## inc86         0.3    2.5  -2.9   0.2   3.5
##
## Fit Diagnostics:
##            mean   sd   10%   50%   90%
## mean_PPD 0.0    0.0  0.0   0.0   0.0
##
## The mean_ppd is the sample average posterior predictive distribution of the outcome variable (for de
##
## MCMC diagnostics
##               mcse Rhat n_eff
## (Intercept)   0.2  1.0   932
## v86_adj       0.3  1.0   871
## inc86         0.1  1.0   798
## mean_PPD      0.0  1.0  3662
## log-posterior 0.0  1.0   906
##
## For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective sample
```

```
pp_check(fit_9a)
```




**(b)**

Fit a robit regression and assess model fit.

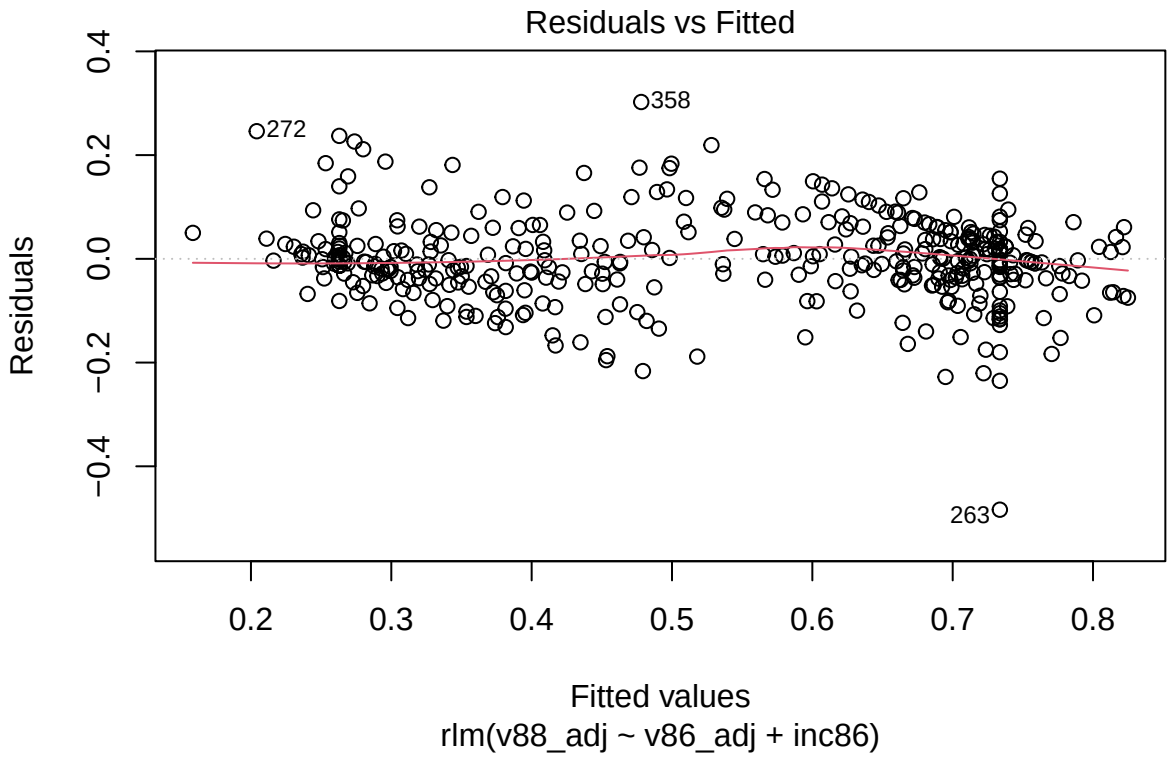
```
fit_9b <- rlm(v88_adj ~ v86_adj + inc86, family = binomial(link = "probit"), data = congress)
```

```
## Warning in rlm.default(x, y, weights, method = method, wt.method = wt.method, :
## some of ... do not match
```

```
summary(fit_9b)
```

```
##
## Call: rlm(formula = v88_adj ~ v86_adj + inc86, data = congress, family = binomial(link = "probit"))
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.48362 -0.03777  0.00102  0.03605  0.30235
##
## Coefficients:
##             Value   Std. Error t value
## (Intercept)  0.0978  0.0193     5.0560
## v86_adj      0.8009  0.0366    21.8972
## inc86        0.0351  0.0077     4.5509
##
## Residual standard error: 0.05597 on 432 degrees of freedom
```

```
plot(fit_9b, which = 1)
```




**(c)**

Which model do you prefer? robit regression model provides better residuals for conparison.

## 15.14 Model checking for count data

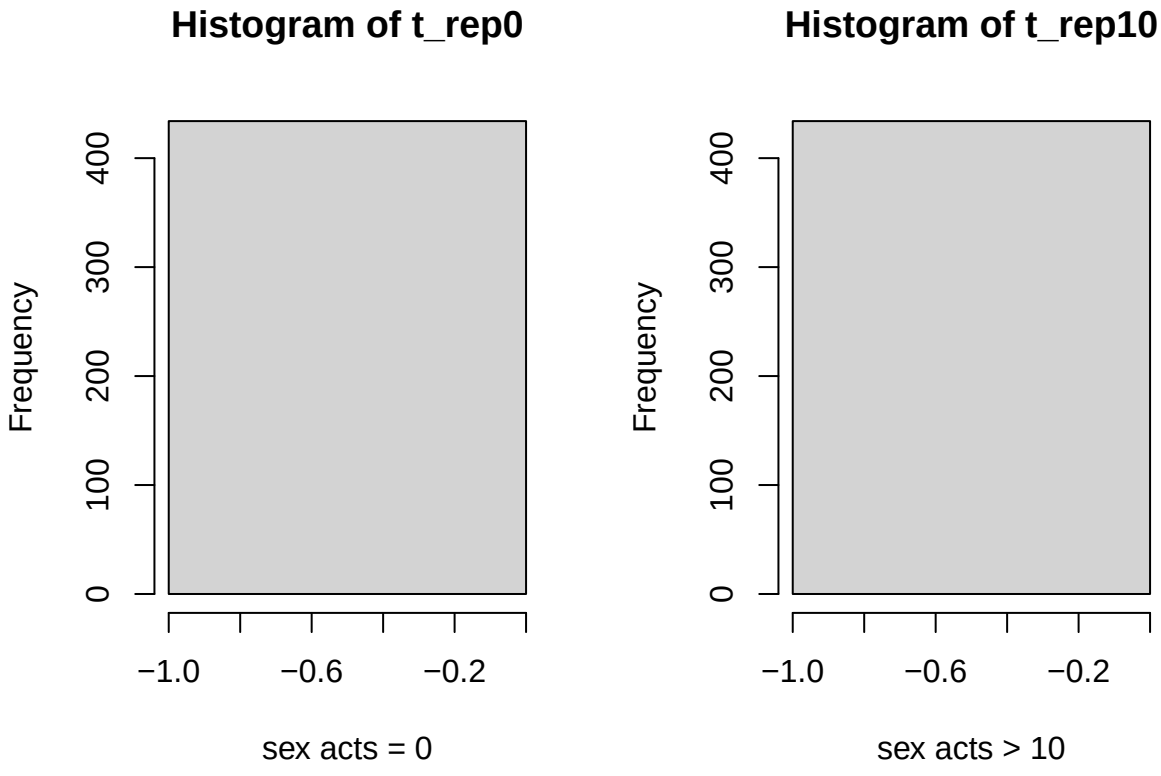
The folder `RiskyBehavior` contains data from a study of behavior of couples at risk for HIV; see Exercise 15.1.

**(a)**

Fit a Poisson regression predicting number of unprotected sex acts from baseline HIV status. Perform predictive simulation to generate 1000 datasets and record the percentage of observations that are equal to 0 and the percentage that are greater than 10 (the third quartile in the observed data) for each. Compare these to the observed value in the original data.

```
fit_15a <- stan_glm(fupacts ~ bs_hiv, family = poisson(link="log"), data = risky, refresh = 0)
risky$bs_hiv_bin <- ifelse(risky$bs_hiv == "negative", 0, 1)


n_sim <- 1000
```

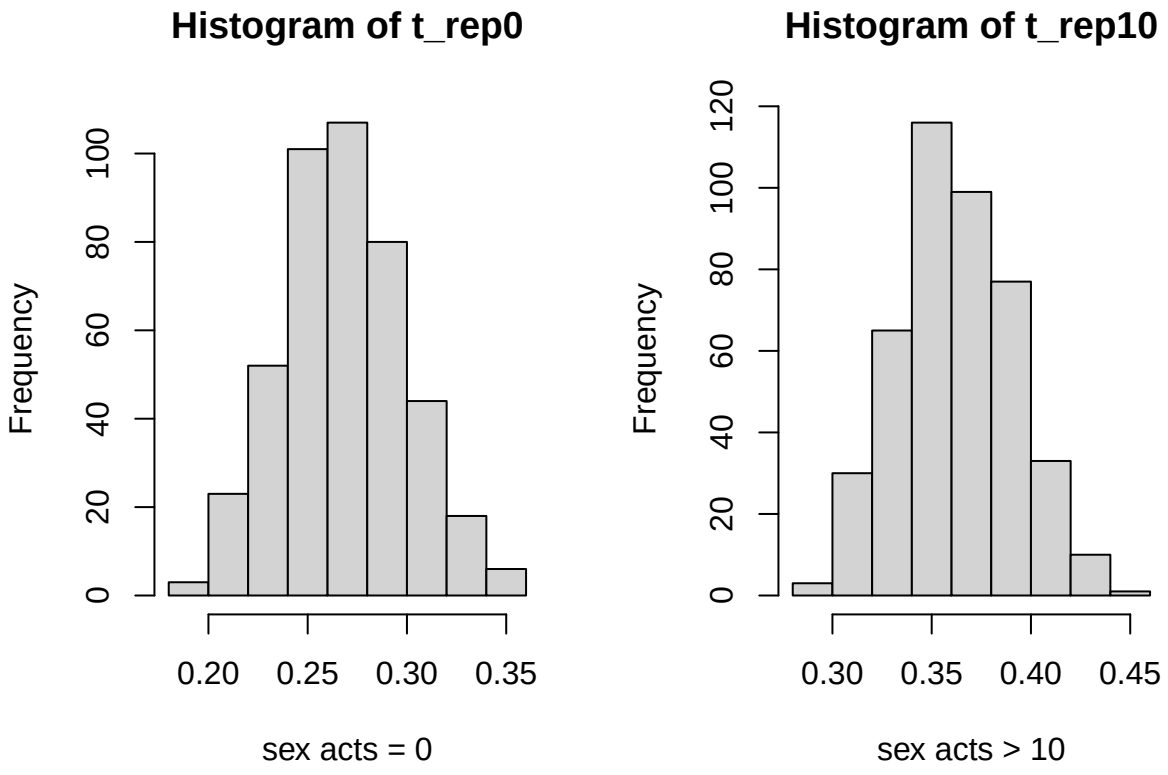
```r
fit_15a1 <- predict(fit_15a, draws = n)
#it keep goes wrong in KNIT but works well if run in RMD
n <- length(risky$fupacts)


t_rep0 <- rep(NA, n_sim)
t_rep10 <- rep(NA, n_sim)
r_0 <- mean(risky$fupacts == 0)
r_10 <- mean(risky$fupacts > 10)

for (i in 1:n) {
  t_rep0[i] <- sum(as.numeric(fit_15a1[i ] == 0))/n
  t_rep10[i] <- sum(as.numeric(fit_15a1[i ] > 10))/n
}

par(mfrow = c(1, 2))
hist(t_rep0, xlab = "sex acts = 0")
hist(t_rep10, xlab = "sex acts > 10")
```




**(b)**

Repeat (a) using a negative binomial (overdispersed Poisson) regression.

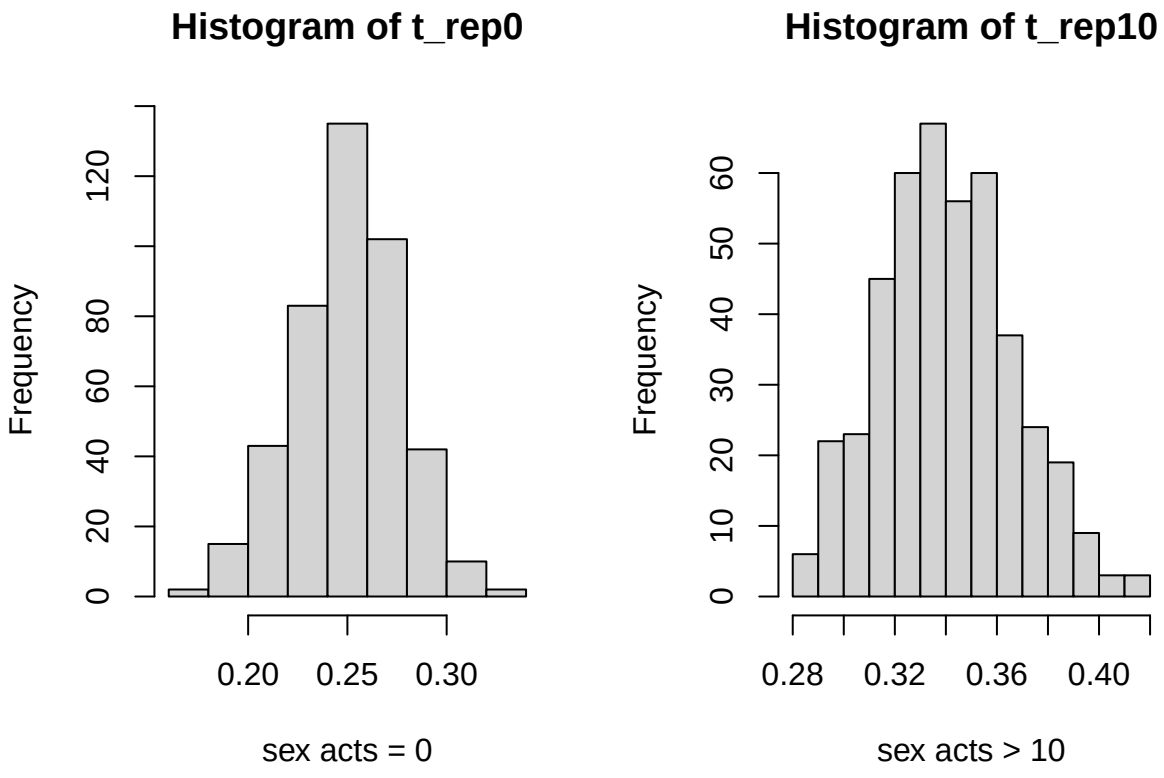
```r
fit_15b <- stan_glm(fupacts ~ bs_hiv, family=neg_binomial_2(link = "log"), data = risky, refresh = 0)


n_sim <- 1000
fit_15b2 <- posterior_predict(fit_15b, draws = n)
n <- length(risky$fupacts)


t_rep0 <- rep(NA, n_sim)
t_rep10 <- rep(NA, n_sim)
r_0 <- mean(risky$fupacts == 0)
r_10 <- mean(risky$fupacts > 10)

for (i in 1:n) {
  t_rep0[i] <- sum(as.numeric(fit_15b2[i, ] == 0))/n
  t_rep10[i] <- sum(as.numeric(fit_15b2[i, ] > 10))/n
}

par(mfrow = c(1, 2))
hist(t_rep0, xlab = "sex acts = 0")
hist(t_rep10, xlab = "sex acts > 10")
```




(c)

Repeat (b), also including ethnicity and baseline number of unprotected sex acts as inputs.

```r
fit_15c <- stan_glm(fupacts ~ bs_hiv + log(bupacts + 1) + sex, data=risky, family=neg_binomial_2(link =



n_sim <- 1000
fit_15c2 <- posterior_predict(fit_15c, draws = n)
n <- length(risky$fupacts)


t_rep0 <- rep(NA, n_sim)
t_rep10 <- rep(NA, n_sim)
r_0 <- mean(risky$fupacts == 0)
r_10 <- mean(risky$fupacts > 10)

for (i in 1:n) {
  t_rep0[i] <- sum(as.numeric(fit_15c2[i, ] == 0))/n
  t_rep10[i] <- sum(as.numeric(fit_15c2[i, ] > 10))/n
}

par(mfrow = c(1, 2))
hist(t_rep0, xlab = "sex acts = 0")
hist(t_rep10, xlab = "sex acts > 10")
```



## 15.15 Summarizing inferences and predictions using simulation

Exercise 15.7 used a Tobit model to fit a regression with an outcome that had mixed discrete and continuous data. In this exercise you will revisit these data and build a two-step model: (1) logistic regression for zero earnings versus positive earnings, and (2) linear regression for level of earnings given earnings are positive. Compare predictions that result from each of these models with each other.

```
lalonde <- haven::read_dta("http://www.nber.org/~rdehejia/data/nsw_dw.dta")
data_1 <- lalonde
data_1$re78 <- ifelse((data_1$re78 > 0), 1, 0)
fit15_1 <- glm(re78 ~ treat + age + education, family = binomial(link = "log"), data = data_1)
print(fit15_1)
```

```
##
## Call:  glm(formula = re78 ~ treat + age + education, family = binomial(link = "log"),
##     data = data_1)
##
## Coefficients:
## (Intercept)        treat          age    education
##   -0.434261     0.157575    -0.000637     0.001361
##
## Degrees of Freedom: 444 Total (i.e. Null);  441 Residual
## Null Deviance:       549.5
## Residual Deviance: 543.1     AIC: 551.1
```

```
data_2 <- lalonde[which(lalonde$re78 > 0), ]
fit15_2 <- glm(re78 ~ treat + age + education, data = data_2)
print(fit15_2)
```

```
##
## Call:  glm(formula = re78 ~ treat + age + education, data = data_2)
##
## Coefficients:
## (Intercept)        treat          age    education
##      114.28      1074.41        69.28       519.70
##
## Degrees of Freedom: 307 Total (i.e. Null);  304 Residual
## Null Deviance:       1.396e+10
## Residual Deviance: 1.345e+10     AIC: 6302
```