Poisson Regression

Drew Davison - STA 310: Homework 4

Instructions:

- Write all narrative using full sentences. Write all interpretations and conclusions in the context of the data.
- Be sure all analysis code is displayed in the rendered pdf.
- If you are fitting a model, display the model output in a neatly formatted table. (The tidy and kable functions can help!)
- If you are creating a plot, use clear and informative labels and titles.
- Make sure to upload to both Gradescope and Canvas in a reproducible format per the instructions of prior homework assignments.

These exercises are derived from BMLR, Chapter 4.

1.

Answer parts a - d in the context of the following study:

A state wildlife biologist collected data from 250 park visitors as they left at the end of their stay. Each was asked to report the number of fish they caught during their one-week stay. On average, visitors caught 21.5 fish per week.

a. Define the response.

The response variable is the number of fish caught per visitor during their one-week stay. This is a discrete and non-negative quantity since it represents the number of fish caught.

b. What are the possible values for the response?

The possible values are non-negative integers starting from 0.

c. What does λ represent?

In a Poisson model, λ represents the mean and the variance of the number of fish caught per visitor. From the given information, λ is 21.5 fish per week.

d. Would a zero-inflated model be considered here? If so, what would be a "true zero"?

If there are park visitors who did not attempt to fish during their stay, then yes, a zero-inflated model would be considered in this scenario. If a significant number of visitors report zero fish caught, then it should be considered.

A "true zero" in this case would refer to a visitor who actively fished but caught no fish. This is different from a structural zero, where a visitor did not attempt to fish at all. In a zero-inflated model, structural zeros are modeled separately from the count process.

2.

@brockmann1996 carried out a study of nesting female horseshoe crabs. Female horseshoe crabs often have male crabs attached to a female's nest known as satellites. One objective of the study was to determine which characteristics of the female were associated with the number of satellites. Of particular interest is the relationship between the width of the female carapace and satellites.

The data can be found in crab.csv in the data folder. It includes the following variables:

- Satellite = number of satellites
- Width = carapace width (cm)
- Weight = weight (kg)
- Spine = spine condition (1 = both good, 2 = one worn or broken, 3 = both worn or broken)
- Color = color (1 = light medium, 2 = medium, 3 = dark medium, 4 = dark)

Make sure to convert Spine and Color to the appropriate data types in R before doing the analysis.

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(tidymodels)
```

```
## -- Attaching packages -------------------------------------- tidymodels 1.2.0 --
## v broom        1.0.6     v rsample      1.2.1
## v dials        1.3.0     v tune         1.2.1
## v infer        1.0.7     v workflows    1.1.4
## v modeldata    1.4.0     v workflowsets 1.1.0
## v parsnip      1.2.1     v yardstick    1.3.1
## v recipes      1.1.0
## -- Conflicts ----------------------------------------- tidymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Dig deeper into tidy modeling with R at https://www.tmwr.org
```

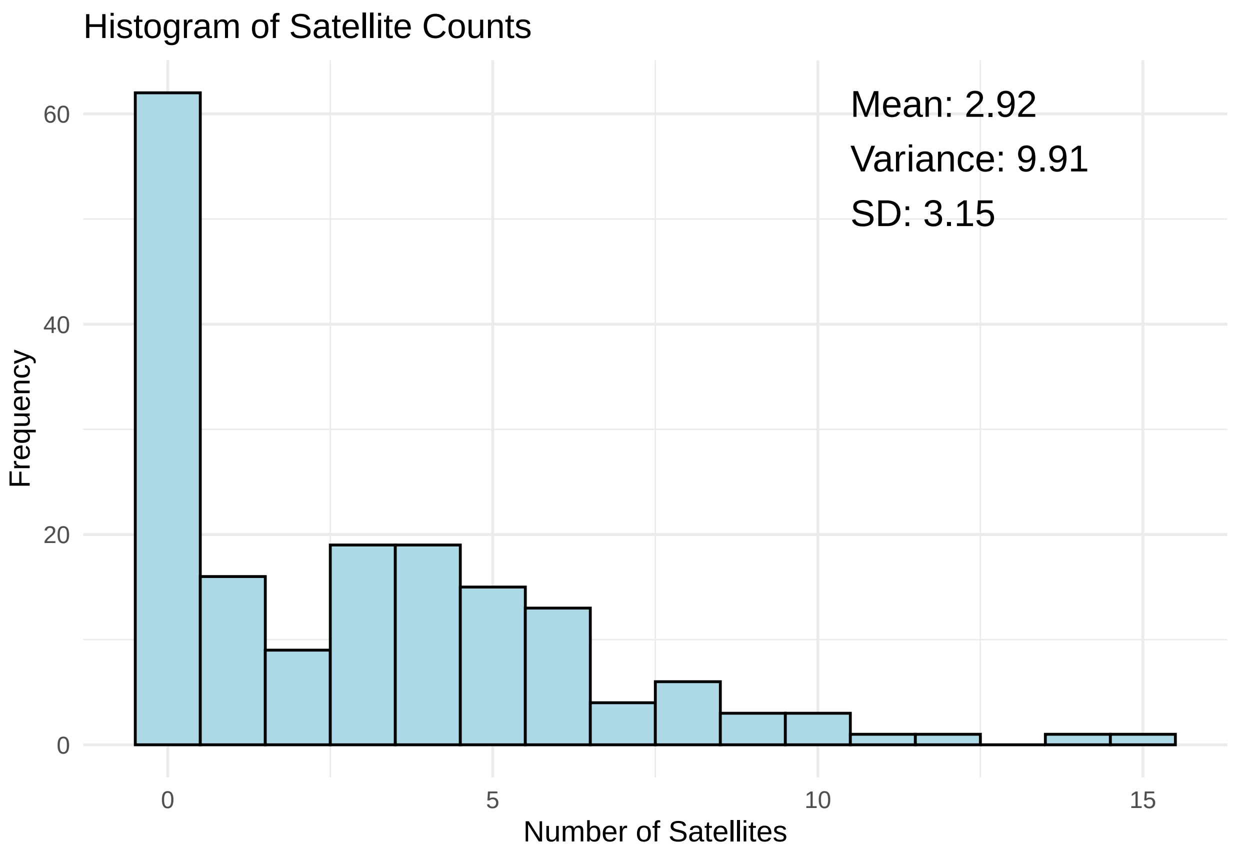
```r
library(skimr)
crab<- read_csv("~/STA310/crab.csv")
```

```
## Rows: 173 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (5): Color, Spine, Width, Satellite, Weight
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```r
crab$Spine <- factor(crab$Spine, levels = c(1, 2, 3),
                   labels = c("Both Good", "One Worn or Broken", "Both Worn or Broken"))

crab$Color <- factor(crab$Color, levels = c(1, 2, 3, 4),
                   labels = c("Light Medium", "Medium", "Dark Medium", "Dark"))
```

a. Create a histogram of Satellite. Is there preliminary evidence the number of satellites could be modeled as a Poisson response? Briefly explain.

```r
library(ggplot2)
library(readr)

sat_mean <- mean(crab$Satellite)
sat_var <- var(crab$Satellite)
sat_sd <- sd(crab$Satellite)

ggplot(crab, aes(x = Satellite)) +
  geom_histogram(binwidth = 1, color = "black", fill = "lightblue") +
  labs(title = "Histogram of Satellite Counts",
       x = "Number of Satellites",
       y = "Frequency") +
  annotate("text", x = max(crab$Satellite) * 0.7, y = max(table(crab$Satellite)) * 0.9,
           label = paste0("Mean: ", round(sat_mean, 2),
                          "\nVariance: ", round(sat_var, 2),
                          "\nSD: ", round(sat_sd, 2)),
           size = 5, hjust = 0) +
  theme_minimal()
```



Yes, there is preliminary evidence for the number of satellites being modeled as a Poisson response. The variable is a non-negative integer beginning at zero, and the distribution of the variable is right-skewed. However, it is worth noting that there is a potential for overdispersion, as the variance is greater than the mean.

b. Fit a Poisson regression model including Width, Weight, and Spine as predictors. Display the model with the 95% confidence interval for each coefficient.

```
poisson_model <- glm(Satellite ~ Width + Weight + Spine,
                     family = poisson(link = "log"), data = crab)

tidy(poisson_model, conf.int = TRUE, exponentiate = TRUE, conf.level = 0.95)
```

```
## # A tibble: 5 x 7
##   term                   estimate std.error statistic p.value conf.low conf.high
##   <chr>                     <dbl>     <dbl>     <dbl>   <dbl>    <dbl>     <dbl>
## 1 (Intercept)               0.346  0.928       -1.14  0.253     0.0564      2.15
## 2 Width                     1.04   0.0477       0.816 0.415     0.946       1.14
## 3 Weight                    1.00   0.000162     2.77  0.00560   1.00        1.00
## 4 SpineOne Worn or Brok~    0.807  0.211       -1.02  0.309     0.525       1.20
## 5 SpineBoth Worn or Bro~    0.952  0.108       -0.458 0.647     0.773       1.18
```

c. Describe the effect of Spine in terms of the mean number of satellites.

Since I exponentiated the coefficients, these values represent rate ratios on the mean number of satellites.

The baseline category of Spine is "Both Good."

The mean number of satellites for crabs with one worn or broken spine is 0.807 times (or ~19.3% lower) than that of crabs with both good spines.

The mean number of satellites for crabs with both spines worn or broken is 0.952 times (or ~4.8% lower) than that of crabs with both good spines.

Crabs with spine damage tend to have fewer satellites than those with both good spines. Greater damage (one vs. both spines worn/broken) does not show a large additional decrease, as going from one worn/broken to both worn/broken only slightly reduces the mean number of satellites.

3. Use the scenario from the previous exercise to answer questions (a) - (d).

a. We would like to fit a quasi-Poisson regression model for this data. Briefly explain why we may want to consider fitting a quasi-Poisson regression model for this data.

We may want to consider fitting a quasi-Poisson regression model due to the overdispersion in Satellite, as mentioned in part 2a. Overdispersion can be solved through a quasi-Poisson regression because it adjusts the standard errors.

b. Fit a quasi-Poisson regression model that corresponds with the model chosen in the previous exercise. Display the model.

```
quasi_poisson_model <- glm(Satellite ~ Width + Weight + Spine,
                               family = quasipoisson(link = "log"), data = crab)

tidy(quasi_poisson_model)
```

```
## # A tibble: 5 x 5
##   term                      estimate std.error statistic p.value
##   <chr>                        <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)              -1.06      1.65        -0.643   0.521
## 2 Width                     0.0389    0.0849       0.458   0.647
## 3 Weight                    0.000450  0.000289     1.56    0.122
## 4 SpineOne Worn or Broken  -0.214     0.375       -0.571   0.568
## 5 SpineBoth Worn or Broken -0.0493    0.192       -0.257   0.797
```

c. What is the estimated dispersion parameter? Show how this value is calculated.

```
dispersion <- summary(quasi_poisson_model)$deviance /
  summary(quasi_poisson_model)$df.residual
dispersion
```

```
## [1] 3.326346
```

$\hat{\phi} = \frac{\sum(\text{Pearson residuals}^2)}{n-p}$

The above is the formula for the estimated dispersion parameter.

Residual Deviance is a measure of the discrepancy between the fitted model and the data. Residual Degrees of Freedom is calculated as the number of observations minus the number of parameters estimated.

If the dispersion parameter is close to 1, the model fits the data similarly to a Poisson model. If it's significantly larger than 1, it indicates overdispersion.

d. How do the estimated coefficients change compared to the model chosen in the previous exercise? How do the standard errors change?

When you move from a Poisson regression model to a quasi-Poisson regression model, the estimated coefficients themselves will generally remain similar. However, the standard errors will likely increase in the quasi-Poisson model, particularly if there is overdispersion in the data. This is because the quasi-Poisson model adjusts for overdispersion by modifying the standard errors.

```
tidy(poisson_model)
```

```
## # A tibble: 5 x 5
##   term                      estimate std.error statistic p.value
##   <chr>                        <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)              -1.06      0.928       -1.14  0.253
## 2 Width                     0.0389    0.0477       0.816 0.415
## 3 Weight                    0.000450  0.000162     2.77  0.00560
## 4 SpineOne Worn or Broken  -0.214     0.211       -1.02  0.309
## 5 SpineBoth Worn or Broken -0.0493    0.108       -0.458 0.647
```

```
tidy(quasi_poisson_model)
```

```
## # A tibble: 5 x 5
##   term                      estimate std.error statistic p.value
##   <chr>                        <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)              -1.06      1.65        -0.643   0.521
## 2 Width                     0.0389    0.0849       0.458   0.647
## 3 Weight                    0.000450  0.000289     1.56    0.122
## 4 SpineOne Worn or Broken  -0.214     0.375       -0.571   0.568
## 5 SpineBoth Worn or Broken -0.0493    0.192       -0.257   0.797
```

In this case, the estimates have remained the same, but the standard errors are much larger in the quasi-Poisson model.

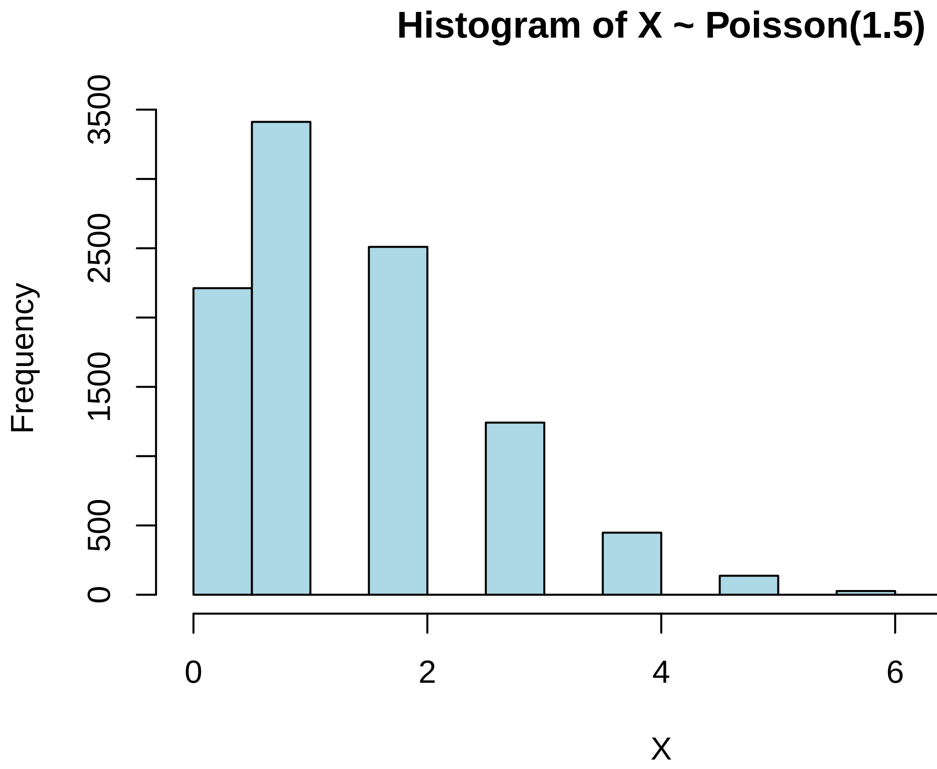
4. The goal of this exercise is to use simulation to understand the equivalency between a gamma-Poisson mixture and a negative binomial distribution.

Remember to set a seed so your simulations are reproducible!

a. Use the R function rpois() to generate 10,000 x_i from a regular Poisson distribution, $X \sim \text{Poisson}(\lambda = 1.5)$. Plot a histogram of this distribution and note its mean and variance. Next, let $Y \sim \text{Gamma}(r = 3, \lambda = 2)$ and use rgamma() to generate 10,000 random y_i from this distribution.

Now, consider 10,000 different Poisson distributions where $\lambda_i = y_i$. Randomly generate one z_i from each Poisson distribution. Plot a histogram of these z_i and compare it to your original histogram of X (where $X \sim \text{Poisson}(1.5)$). How do the means and variances compare?

```
set.seed(123)

X <- rpois(10000, lambda = 1.5)

hist(X, main = "Histogram of X ~ Poisson(1.5)",
     xlab = "X", col = "lightblue", border = "black")
```



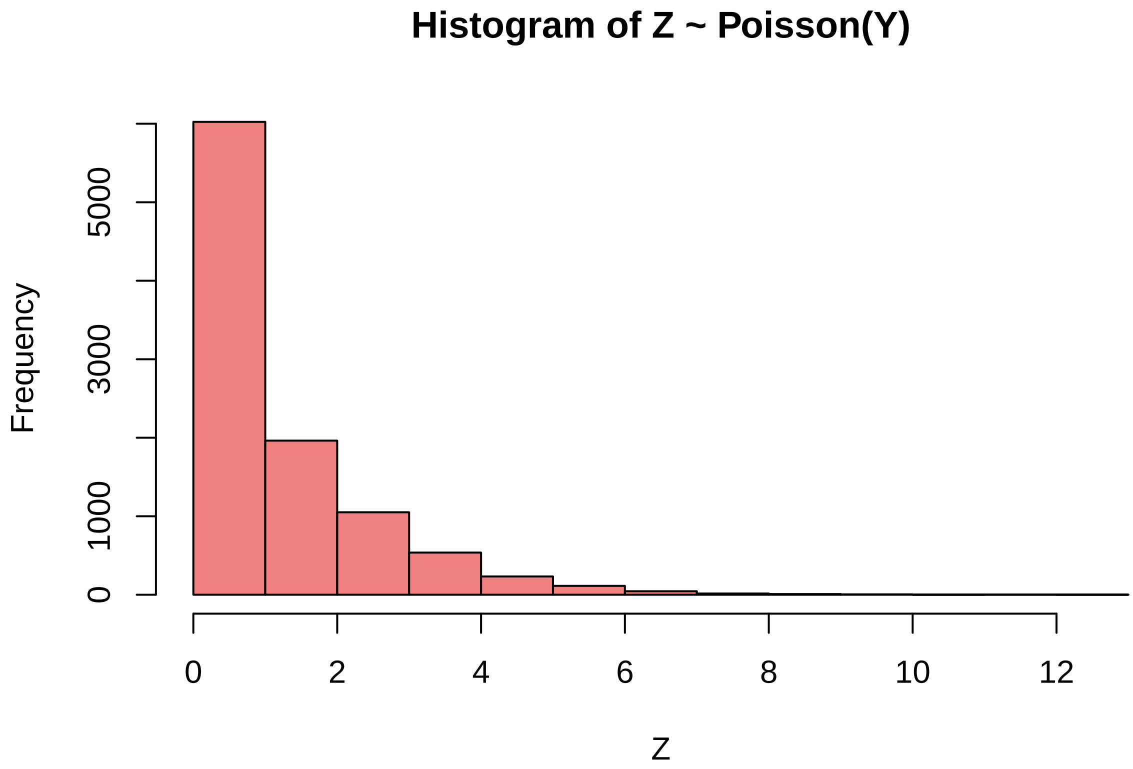
```
mean_X <- mean(X)
var_X <- var(X)

cat("Mean of X: ", mean_X, "\n")
```

```
## Mean of X:  1.4882
```

```
cat("Variance of X: ", var_X, "\n")
```

```
## Variance of X:  1.465207
```

```
set.seed(123)

Y <- rgamma(10000, shape = 3, rate = 2)

Z <- sapply(Y, function(lambda) rpois(1, lambda))

hist(Z, main = "Histogram of Z ~ Poisson(Y)",
     xlab = "Z", col = "lightcoral", border = "black")
```



```
mean_Z <- mean(Z)
var_Z <- var(Z)

cat("Mean of Z: ", mean_Z, "\n")
```

```
## Mean of Z:  1.4635
```

```
cat("Variance of Z: ", var_Z, "\n")
```

```
## Variance of Z:  2.235891
```

The histograms are very different as the first has gaps and a mode larger than in the second.

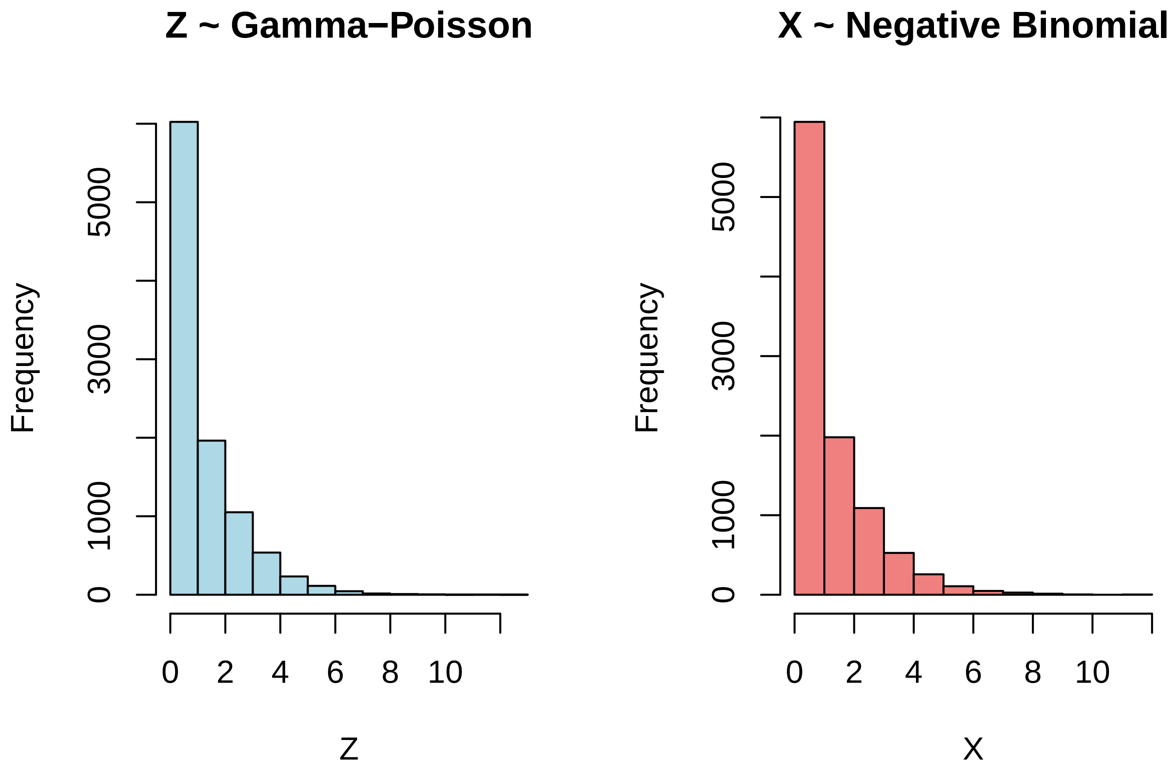
The means of the two are similar, but the variance in the second is greater.

b. A negative binomial distribution can actually be expressed as a gamma-Poisson mixture. In Part a, you looked at a gamma-Poisson mixture $Z \sim \text{Poisson}(\lambda)$ where $\lambda \sim \text{Gamma}(r = 3, \lambda' = 2)$.

Find the parameters of a negative binomial distribution $X \sim \text{NegBinom}(r, p)$ such that X is equivalent to Z. As a hint, the means of both distributions must be the same, so $\frac{r(1-p)}{p} = \frac{3}{2}$.

Show through histograms and summary statistics that your negative binomial distribution is equivalent to the gamma-Poisson mixture. You can use rnbinom() in R.

```r
set.seed(123)


r_gamma <- 3
lambda_gamma <- 2
Y <- rgamma(10000, shape = r_gamma, rate = lambda_gamma)
Z <- sapply(Y, function(lambda) rpois(1, lambda))


r_nb <- 3
mean_gamma_poisson <- 3 / 2


p <- r_nb / (r_nb + mean_gamma_poisson)


X <- rnbinom(10000, size = r_nb, prob = p)


par(mfrow = c(1, 2))


hist(Z, main = "Z ~ Gamma-Poisson",
     xlab = "Z", col = "lightblue", border = "black")


hist(X, main = "X ~ Negative Binomial",
     xlab = "X", col = "lightcoral", border = "black")
```



```r
cat("Summary of Gamma-Poisson mixture (Z):\n")
```

```
## Summary of Gamma-Poisson mixture (Z):
```

```r
cat("Mean: ", mean(Z), "\n")
```

```
## Mean:  1.4635
```

```r
cat("Variance: ", var(Z), "\n\n")
```

```
## Variance:  2.235891
```

```r
cat("Summary of Negative Binomial (X):\n")
```

```
## Summary of Negative Binomial (X):
```

```r
cat("Mean: ", mean(X), "\n")
```

```
## Mean:  1.4957
```

```r
cat("Variance: ", var(X), "\n")
```

```
## Variance:  2.297011
```

c. Make an argument that if you want a NegBinom(r, p) random variable, you can instead sample from a Poisson distribution, where the λ values are themselves sampled from a gamma distribution with parameters r and $\lambda' = \frac{p}{1-p}$. You may show equivalency via the simulations or mathematically, however, make sure your arguments are precise and clear.

Negative Binomial Distribution: A random variable $X \sim \text{NegBinom}(r, p)$ has the following characteristics:

- **Mean:** $\mathbb{E}[X] = \frac{r(1-p)}{p}$
- **Variance:** $\text{Var}(X) = \frac{r(1-p)}{p^2}$

The Negative Binomial distribution can be interpreted as the number of failures before a fixed number of successes in a sequence of Bernoulli trials, where the number of trials is random and follows a **Poisson** distribution with parameter λ.

Poisson-Gamma Mixture: A random variable Z is said to follow a Poisson-Gamma mixture if:

- First, we sample a random variable λ from a **Gamma** distribution: $\lambda \sim \text{Gamma}(r, \lambda')$.
- Next, given λ, we sample a Poisson random variable: $Z|\lambda \sim \text{Poisson}(\lambda)$.

This means that the random variable Z is a mixture of Poisson distributions, where the rate parameter λ follows a Gamma distribution.

The **Gamma** distribution with shape parameter r and rate parameter λ' has the following characteristics:

- **Mean:** $\mathbb{E}[\lambda] = \frac{r}{\lambda'}$
- **Variance:** $\text{Var}(\lambda) = \frac{r}{\lambda'^2}$

The **Poisson** distribution with rate parameter λ has the following characteristics:

- **Mean:** $\mathbb{E}[Z|\lambda] = \lambda$
- **Variance:** $\text{Var}(Z|\lambda) = \lambda$

The **overall mean and variance** of the **Poisson-Gamma mixture** (denoted Z) can be computed as follows:

Mean:

$$\mathbb{E}[Z] = \mathbb{E}[\mathbb{E}[Z|\lambda]] = \mathbb{E}[\lambda] = \frac{r}{\lambda'}$$

Variance:

$$\text{Var}(Z) = \mathbb{E}[\text{Var}(Z|\lambda)] + \text{Var}(\mathbb{E}[Z|\lambda]) = \mathbb{E}[\lambda] + \text{Var}(\lambda) = \frac{r}{\lambda'} + \frac{r}{\lambda'^2}$$

Thus, the variance of the Gamma-Poisson mixture is:

$$\text{Var}(Z) = \frac{r(1 + \lambda')}{\lambda'^2}$$

Now, let's examine the **Negative Binomial** distribution:

Mean: For $X \sim \text{NegBinom}(r, p)$, we have:

$$\mathbb{E}[X] = \frac{r(1-p)}{p}$$

Variance:

$$\text{Var}(X) = \frac{r(1-p)}{p^2}$$

To connect the two distributions, we look for parameters of the Gamma distribution such that the **mean** and **variance** of the **Gamma-Poisson mixture** match the **mean** and **variance** of the **Negative Binomial distribution**.

From the equivalency of the means:

$$\frac{r}{\lambda'} = \frac{r(1-p)}{p}$$

This leads to:

$$\lambda' = \frac{p}{1-p}$$

Thus, the rate parameter λ' of the Gamma distribution should be $\frac{p}{1-p}$ for the **Gamma-Poisson mixture** to match the **Negative Binomial** distribution.

5. In a 2018 study, Chapp et al. (2018) scraped every issue statement from webpages of candidates for the U.S. House of Representatives, counting the number of issues candidates commented on and scoring the level of ambiguity of each statement. We will focus on the issue counts, and determining which attributes (of both the district as a whole and the candidates themselves) are associated with candidate silence (commenting on 0 issues) and a willingness to comment on a greater number of issues. The data set is in ambiguity.csv in the data folder . This analysis will focus on the following variables:

- name : candidate name
- distID : unique identification number for Congressional district
- ideology : candidate left-right orientation
- democrat : 1 if Democrat, 0 if Republican
- totalIssuePages : number of issues candidates commented on (response)

See @roback2021beyond for the full list of variables.

We will use a **hurdle model** to analyze the data. A hurdle model is similar to a zero-inflated Poisson model, but instead of assuming that "zeros" are comprised of two distinct groups—those who would always be 0 and those who happen to be 0 on this occasion—the hurdle model assumes that "zeros" are a single entity. Therefore, in a hurdle model, cases are classified as either "zeros" or "non-zeros", where "non-zeros" *hurdle* the 0 threshold—they must always have counts of 1 or above.

We will use the `pscl` package and the `hurdle` function in it to analyze a hurdle model. Note that coefficients in the "zero hurdle model" section of the output relate predictors to the log-odds of being a *non-zero* (i.e., having at least one issue statement), which is opposite of the ZIP model.

a. Visualize the distribution of the response variable totalIssuePages. Why might we consider using a hurdle model compared to a Poisson model? Why is a zero-inflated Poisson model not appropriate in this scenario?

b. Create a plot of the empirical log odds of having at least one issue statement by ideology. You may want to group ideology values first. What can you conclude from this plot?

c. Create a hurdle model with ideology and democrat as predictors in both parts. Display the model. Interpret ideology in both parts of the model.

d. Repeat (d), but include an interaction in both parts. Interpret the interaction in the zero hurdle part of the model.

Grading

Total	**39**
Ex 1	4
Ex 2	6
Ex 3	8
Ex 4	8
Ex 5	10
Workflow & formatting	3

The "Workflow & formatting" grade is to based on the organization of the assignment write up along with the reproducible workflow. This includes having an organized write up with neat and readable headers, code, and narrative, including properly rendered mathematical notation. It also includes having a reproducible .Rmd document that can be rendered to reproduce the submitted PDF.

Extra resources and hints for Exercise 5 (Hurdle Problem)

1. There is an article on hurdle models that has helped some students that can be found here:

https://jsdajournal.springeropen.com/articles/10.1186/s40488-021-00121-4

2. There are two parts to the hurdle model, the count part and the binary part of the model and how to run it can be found in the R documentation.
3. You are not required to account for overdispersion (or check for it), however, if you do, this is great and just please make sure to think through how to do this properly as it involves integrating multiple parts of the Poisson lectures.
4. How do we handle the data in 5b? Do we bin it?

Yes, you should bin it or group it using the function cut_interval.

For example, something like this might help:

ideologybin = cut_interval(ideology, n=5)), where the number of bins of 5 was chosen empirically by playing around with the data.

References