Homework 4

Denis Ostroushko

Problem 1

(1)

First, we will fit the two model and display the code so that you may follow along:

```
f1 <- glm(seiz ~ log(age)+log_base+treat, data = seiz_total, family = poisson())
fq1 <- glm(seiz ~ log(age)+log_base+treat, data = seiz_total, family = quasipoisson())
```

Using the diagonal of a variance-covariance matrix we can obtain variances of each model coefficient.

Variances of fitted coefficients from the Poisson regression model are given below:

```
diag(vcov(f1))
```

```
(Intercept)    log(age)    log_base       treat
0.162843551 0.012083084 0.001057737 0.002321266
```

Variances of fitted coefficients from the Quasipoisson regression model are given below:

```
diag(vcov(fq1))
```

```
(Intercept)    log(age)    log_base       treat
 1.80021599  0.13357705  0.01169316  0.02566131
```

Taking the ratio of the two coefficient vectors we obtain a constant value of the over-dispersion parameter

```
diag(vcov(fq1)) / diag(vcov(f1))
```

```
(Intercept)    log(age)    log_base       treat 
   11.05488    11.05488    11.05488    11.05488 
```

Note that the summary output of the Quasipoisson regression model provides the same value:

```

Call:
glm(formula = seiz ~ log(age) + log_base + treat, family = quasipoisson(), 
    data = seiz_total)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-6.0834  -2.0602  -0.4096   1.3963   8.1997  

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.02151    1.34172  -0.761    0.450    
log(age)     0.58778    0.36548   1.608    0.114    
log_base     1.22522    0.10813  11.330 5.27e-16 ***
treat       -0.01759    0.16019  -0.110    0.913    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 11.05488)

    Null deviance: 2122.73  on 58  degrees of freedom
Residual deviance:  556.39  on 55  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5
```

(2)

Using code below with `N_iter` iterations we will obtain bootstrapp sampling distributions of Poisson regression model coefficients.

```
results <-
  data.frame(
    i = seq(1,N_iter, by = 1),
    intercept_b = rep(NA, N_iter),
    log_age_b = rep(NA, N_iter),
    log_base_b = rep(NA, N_iter),
    treat_b = rep(NA, N_iter)
  )

set.seed(1)

for(i in 1:N_iter){

  temp_m <- glm(seiz ~ log(age)+log_base+treat,
                data = seiz_total[sample(rownames(seiz_total), replace= T), ],
                family = poisson())

  results$intercept_b[i] = coef(temp_m)[1]
  results$log_age_b[i] = coef(temp_m)[2]
  results$log_base_b[i] = coef(temp_m)[3]
  results$treat_b[i] = coef(temp_m)[4]

}
```

Table 1 shows comparison of standard errors for model term coefficients obtained using three different ways.

Table 1: Comparion of fitted and boostrapped model parameters

	Poisson Fitted Values		Quasipoisson Fitted Values		Boostrapp Simple Model	
Model Terms	Beta	SE	Beta	SE	Avg. Beta	SE
Log(Age)	0.59	0.11	0.59	0.37	0.61	0.31
Log(Baseline Seizures)	1.23	0.03	1.23	0.11	1.18	0.17
Treatment	-0.02	0.05	-0.02	0.16	-0.08	0.21

Comments:

1. We can see full effect of Poisson regression deficiency, standard errors for beta's are much much smaller for Possion beta's when compared to Quasipoisson.
2. Quasipoisson and Bootstrap standard errors are *approximately* equal. For some variables

bootstrap standard error is bigger, for some quasipoisson standard errors are higher.

3. Overall, Quasipoisson and Boostrap methods produce similar standard errors, so either method should be appropriate for use in practice.

(3)

Bias = average Beta’s from rN_iter‘ bootstrap iterations - fitted coefficients from regression model.

Summarize bootstrap iterations data and present results:

```
boot_beta <-
  results %>%
    summarise(across(.cols = c("intercept_b", "log_age_b", "log_base_b", "treat_b"), .fns

boot_beta
```

```
  intercept_b log_age_b log_base_b     treat_b
1  -0.9737166 0.6078214   1.175106 -0.07564639
```

True coefficients from the regression model:

```
(Intercept)    log(age)    log_base       treat
-1.02151177  0.58777978  1.22521697 -0.01759081
```

Taking the difference produces bias for each model coefficient:

```
boot_beta - coef(f1)
```

```
  intercept_b log_age_b  log_base_b     treat_b
1  0.04779522 0.0200416 -0.05011093 -0.05805557
```

Numerically, bias estimates are very close to 0 for each model term, so we have no reason to believe that the estimates are biased.

(4)

We will use function displayed below to obtain normal approximation confidence intervals:

```
normal_approx <-
  function(coef, boot_se, rounding = 5){

    vec = c(
      round(coef - qt(p = 0.975, df = nrow(seiz_total) - 1) * boot_se , rounding),
      round(coef + qt(p = 0.975, df = nrow(seiz_total) - 1) * boot_se , rounding)
    )

   names(vec) <- c("2.5%", "97.5%")

   return(vec)
  }
```

I will use one page per variable in this and future sections for organization purposes.

Log (Age)

Figure 1 shows bootstrap sampling distribution for log(age) predictor from the simple additive model. We use mean and variance of bootstrap sampling distribution to fit and display a normal curve.

Visually, log(age) distribution looks approximately normal.

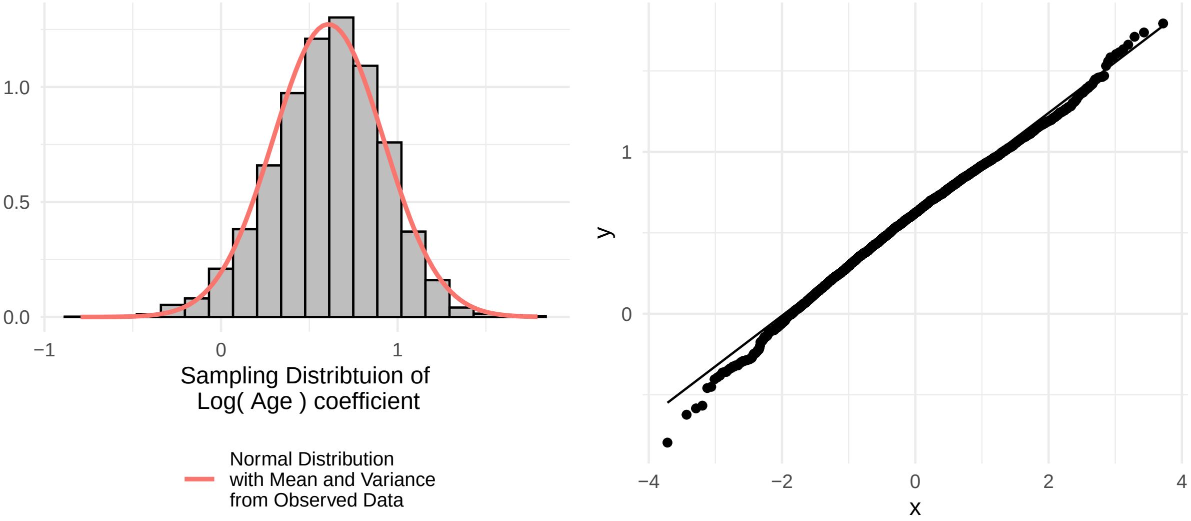

Figure 1: Sampling Distribution of Log(Age) regression coefficient

1. Normal Approximation Method

```
   2.5%    97.5% 
-0.03967  1.21523 
```

2. Percentile Method

```
       2.5%       97.5% 
-0.04301019  1.18145650 
```

3. Comparison with Quasipoisson

```
     2.5 %     97.5 % 
-0.1307821  1.3032966 
```

4. Comments:

- We can see that Quasipoisson produces the widest itnerval in the set of three options. However, it is well within a range that we might want to consider as noise, without conducting more tests to verify the difference.

- Normal approximation is slightly wider than the quantile interval method. We can see that the normal curve and historgram both show slight evidence of heavy tails or values that are starting to be a little extreme. Greater width of distribution increases variance, and therefore we have a higher standard error for the normal approximation confidence interval.

Log (Baseline Seizures)

Figure 2 shows bootstrap sampling distribution for log(age) predictor from the simple additive model.

Visually, log(age) distribution does not look approximately normal. In fact, it looks more bimodal. QQplot also shows evidence of deviation from normality for the sampling distribution.

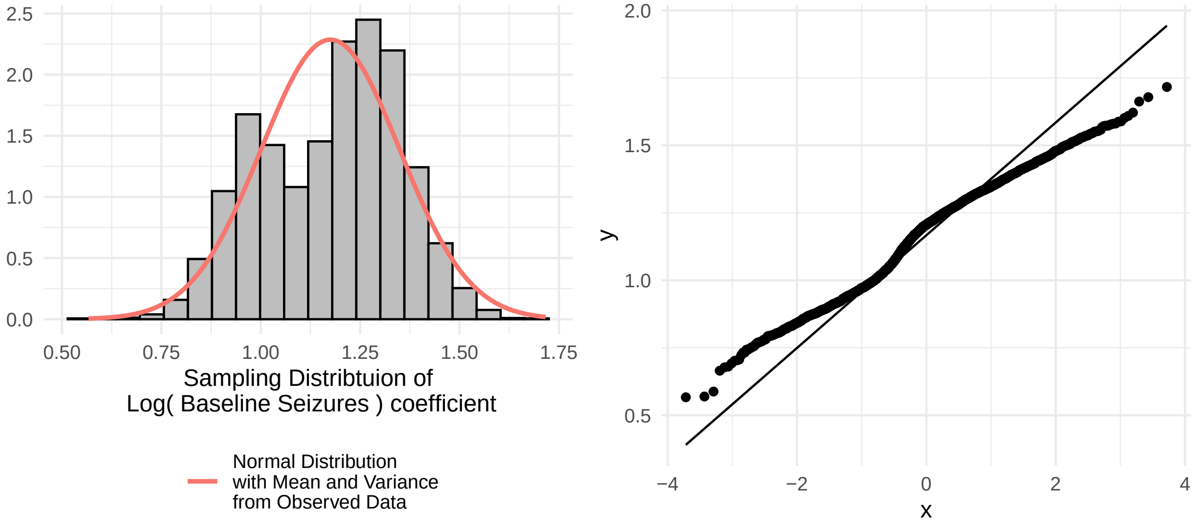

Figure 2: Sampling Distribution of Log(Baseline Seizures) regression coefficient

1. Normal Approximation Method

```
   2.5%   97.5%
0.87600 1.57443
```

2. Percentile Method

```
     2.5%     97.5%
0.8476414 1.4723280
```

3. Comparison with Quasipoisson

```
   2.5 %   97.5 %
1.015432 1.439530
```

4. Comments:

- All three methods produce approximately similar confidence intervals. However, due to deviation from normality, we can no longer interpret these confidence intervals in a regular way.

Treatment

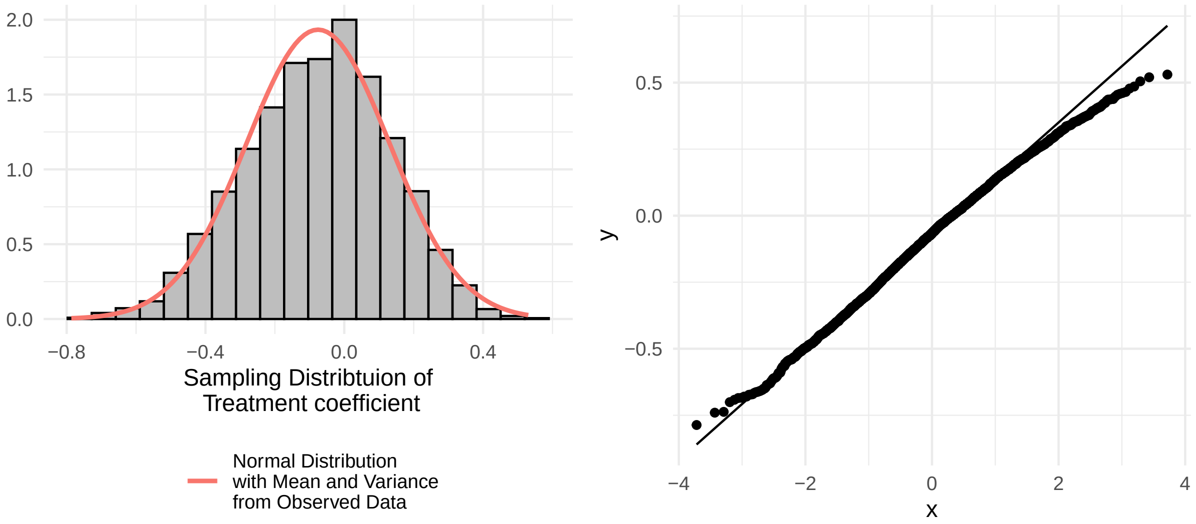

Figure 3: Sampling Distribution of Treatment regression coefficient

1. Normal Approximation Method

```
    2.5%    97.5% 
-0.43063  0.39545 
```

2. Percentile Method

```
      2.5%      97.5% 
-0.4890726  0.3019872 
```

3. Comparison with Quasipoisson

```
     2.5 %     97.5 % 
-0.3318274  0.2970761 
```

4. Comments:

- Normal approximation confidence interval is quite wider than the other two methods. We can see that both tails are quite heavier than expected under the normal distribution. Therefore, variance of the estimate and standard error appear more inflated. Quantiles are less sensitive to more extreme values, therefore, quantile method aligns with the model estiamted confidence interval better.

(5)

Repeat a bootstrap procedure with a model containing a total of eight predictors now.

```r
results_int <-
  data.frame(
    i = seq(1,N_iter, by = 1),
    intercept_b = rep(NA, N_iter),
    log_age_b = rep(NA, N_iter),
    log_base_b = rep(NA, N_iter),
    treat_b = rep(NA, N_iter),
    log_age_log_base_int_b = rep(NA, N_iter),
    log_age_treat_int_b = rep(NA, N_iter),
    log_base_treat_b = rep(NA, N_iter),
    log_age_log_base_treat_int_b = rep(NA, N_iter)

  )

set.seed(1)

for(i in 1:N_iter){

  temp_m <- glm(seiz ~log(age)*log_base*treat,
                data = seiz_total[sample(rownames(seiz_total), replace= T), ],
                family = poisson())

  results_int$intercept_b[i] = coef(temp_m)[1]
  results_int$log_age_b[i] = coef(temp_m)[2]
  results_int$log_base_b[i] = coef(temp_m)[3]
  results_int$treat_b[i] = coef(temp_m)[4]
  results_int$log_age_log_base_int_b[i] = coef(temp_m)[5]
  results_int$log_age_treat_int_b[i] = coef(temp_m)[6]
  results_int$log_base_treat_b[i] = coef(temp_m)[7]
  results_int$log_age_log_base_treat_int_b[i] = coef(temp_m)[8]

}
```

Comparison with simpler additive model

Table 2 displays fitted coefficients and standard errors for a more complicated model, as well as a simpler poisson regression.

Comments:

Table 2: Comparion of fitted and boostrapped model parameters

	Poisson Fitted Values		Quasipoisson Fitted Values		Bootsrapp Model with Interaction		Boostrapp Simple Model	
Model Terms	Beta	SE	Beta	SE	Avg. Beta	SE	Avg. Beta	SE
Log(Age)	-1.39	0.56	-1.39	1.79	-1.49	1.01	0.61	0.31
Log(Baseline Seizures)	-2.44	0.82	-2.44	2.62	-2.58	1.83	1.18	0.17
Treatment	-8.29	2.77	-8.29	8.83	-6.77	7.16	-0.08	0.21
Log(Age):Log(Baseline Seizures)	1.01	0.24	1.01	0.78	1.05	0.55	NA	NA
Log(Age):Treatment	2.05	0.84	2.05	2.69	1.68	2.11	NA	NA
Log(Baseline Seizures_:Treatment	3.53	1.21	3.53	3.84	2.96	3.46	NA	NA
Log(Age):Log(Baseline Seizures):Treatment	-0.87	0.37	-0.87	1.18	-0.74	1.04	NA	NA

- We obviously know that a Poisson regression model will severely underestimate variance. For some coefficients the difference is absolutely enormous.

- This example showcases how bootstrap standard error estimates are consistently lower than the quasipoisson estimates. Jared pointed out that underestimation of standard errors is a known flaw of bootstrap methods.

- It was surprising for me to see that even after 5000 replications, we get a notable discrepancy between quasipoisson regression coefficients and average Beta's from the replications. I am not sure if this suggests that the model estimates are biased. Mathematically, we surely will see a notable amount of bias.

Log (Age)

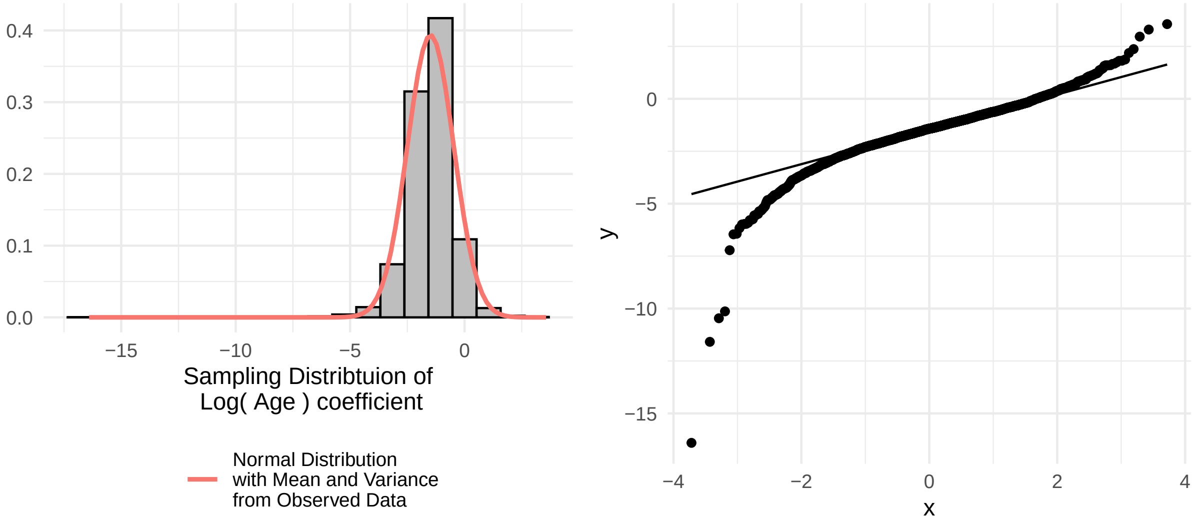

Figure 4: Sampling Distribution of Log(Age) regression coefficient from a model with interaction terms

1. Normal Approximation Method

```
    2.5%    97.5%
-3.41674  0.64253
```

2. Percentile Method

```
      2.5%      97.5%
-3.5617693  0.3128503
```

3. Comparison with Quasipoisson

```
    2.5 %    97.5 %
-4.812707  2.196485
```

4. Comments:

- I wouldn't particularly call this sampling distribution approximately normal. Lower tail is quite heavy, so normal and quantile approximations of the confidence interval might not make sense anymore. Complicated models commonly have such issues and deviations from assumptions.
- Normal approximation method is heavily affected by the variance inflation from lower tail outliers.

- Quantile method is affected by deviation from normality: observed quantiles of the data do not align with theoretical quantiles of a normal distribution

5. Differences:

- The distribution has a heavy hail now, which we did not see on Figure 1.
- Confidence interval endpoints are now quite different as well. Skewness of the distribution affects estiamtes.

Log (Baseline Seizures)

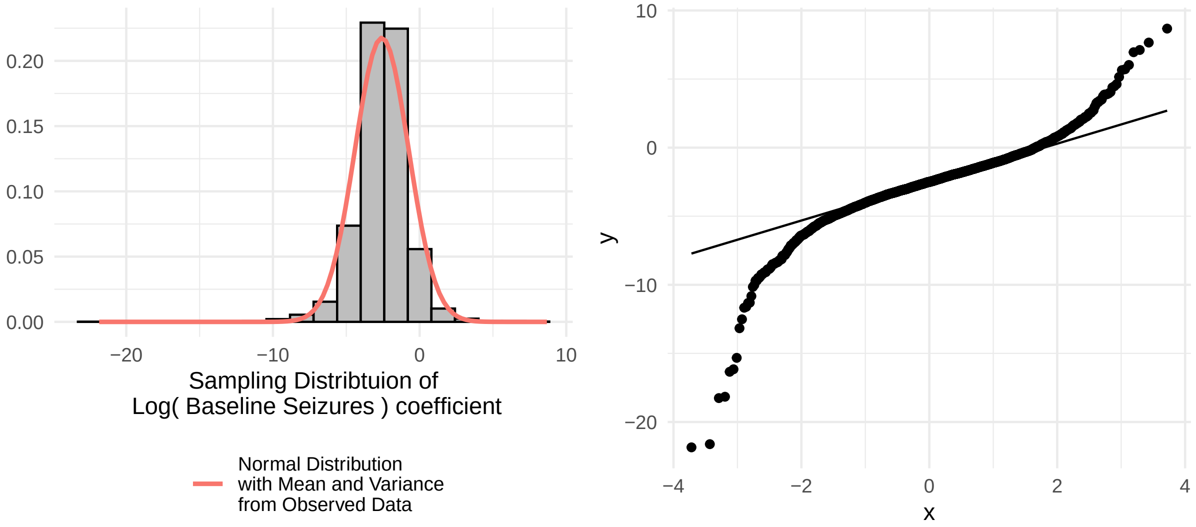

Figure 5: Sampling Distribution of Log(Base) regression coefficient from a model with interaction terms

1. Normal Approximation Method

```
    2.5%    97.5%
-6.10475  1.23402
```

2. Percentile Method C.I.

```
      2.5%      97.5%
-6.3278337  0.7259339
```

3. Comparison with Quasipoisson

```
    2.5 %    97.5 %
-7.471634  2.791414
```

4. Comments and Comparisons:

- More complicated model and interaction of factors changed the distribution of log-baseline seizures. The distribution is no longer bimodal.
- However, I would say there are still issues with the sampling distribution. Heavy tails cause the discrepancy between the three methods we use to obtain confidence intervals.

Treatment

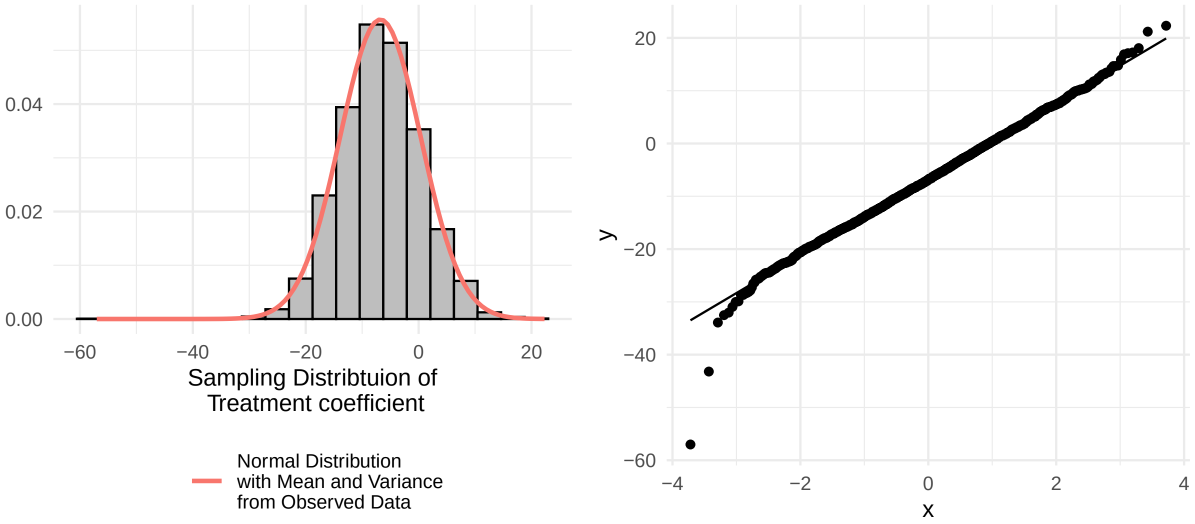

Figure 6: Sampling Distribution of Treatment regression coefficient from a model with interaction terms

1. Normal Approximation Method

```
     2.5%      97.5% 
-22.61895   6.03964 
```

2. Percentile Method C.I.

```
      2.5%       97.5% 
-20.268813   7.213944 
```

3. Comparison with Quasipoisson

```
     2.5 %     97.5 % 
-25.221737   9.446769 
```

4. Comments:

- Treatment flag sampling distribution looks approximately normal, with a minor issue at the lower tail
- It seems to me that the range of normal approximation and percentile methods are very similar, however, the bounds are quite different. I suspect that this goes back to the bias comment I made easrlier. Normal approximation uses a true fitted value of a coefficient

and bootstrap standard error to make an interval. Quantile method relies purely on the estimated sampling distribution, which has a different average - center - which is a the average Beta from all replications. The difference in what we consider the center of the interval causes differing bounds between the two methods

- Quasipoisson method produces a wider confidence interval, and it is wider equally in each direction.

5. Comparisons:

- Everything is different for treatment now. Fitted coefficient is quite large and far from zero
- Confidence interval is also extremely wide when compared with he previous results from a simpler model
- It is hard to say what exactly is the cause of such a large discrepancy without examining how the model behavior changes with addition of each new variable and interaction.

Log(Age) - Log(Baseline Seizures) Interaction Term

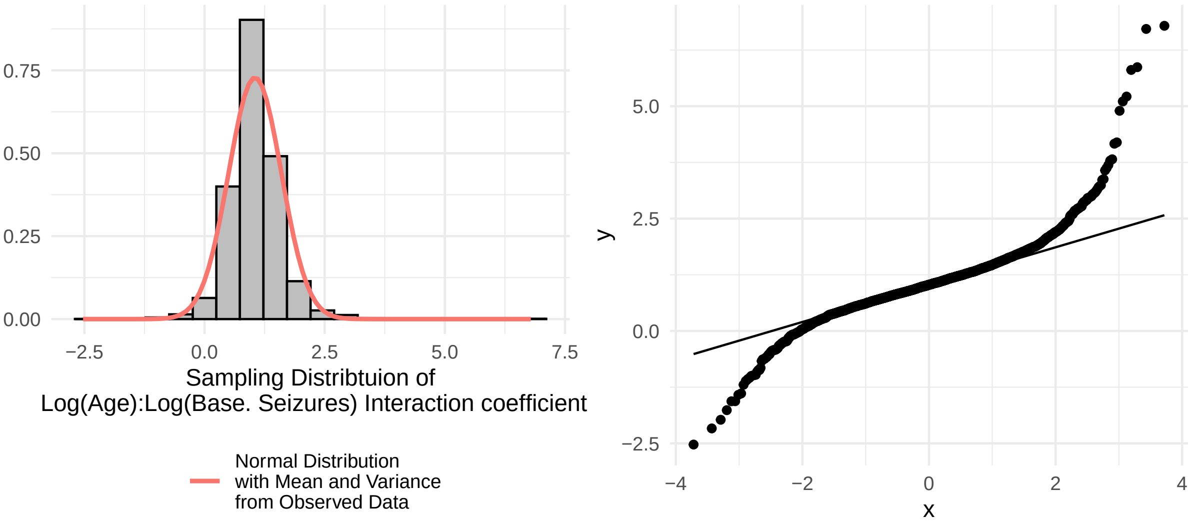

Figure 7: Sampling Distribution of respective interactive term regression coefficient

1. Normal Approximation Method

```
   2.5%    97.5%
-0.09037  2.10347
```

2. Percentile Method C.I.

```
      2.5%      97.5%
0.07128832 2.15557874
```

3. Comparison with Quasipoisson

```
     2.5 %     97.5 %
-0.5424838  2.5149233
```

4. Comments:

- Similar issue - heavy tails of sampling distribution.
- Despite the issue we have confidence interval endpoints that are quite similar. We saw that this is not always the case form previous examinations, so we will treat this occurrence as a coincidence.

Treatment - Log(Age) Interaction Term

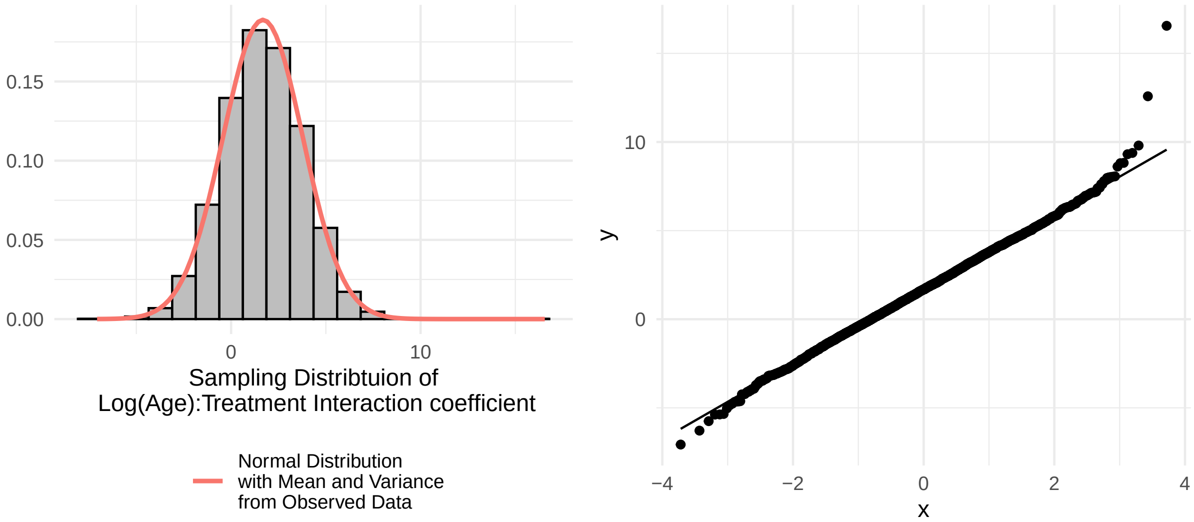

Figure 8: Sampling Distribution of respective interactive term regression coefficient

1. Normal Approximation Method

```
    2.5%     97.5%
-2.17655  6.27760
```

2. Percentile Method C.I.

```
     2.5%      97.5%
-2.497497  5.748052
```

3. Comparison with Quasipoisson

```
    2.5 %     97.5 %
-3.354097  7.201258
```

4. Comments:

- We can see that there is an issue at the upper tail, the tail is quite heavier than expected.
- This issue causes the discrepancy between quantile and normal approximation methods.
- Quasipoisson confidence interval is wider in each direction again.

Treatment - Log(Baseline Seizures) Interaction Term

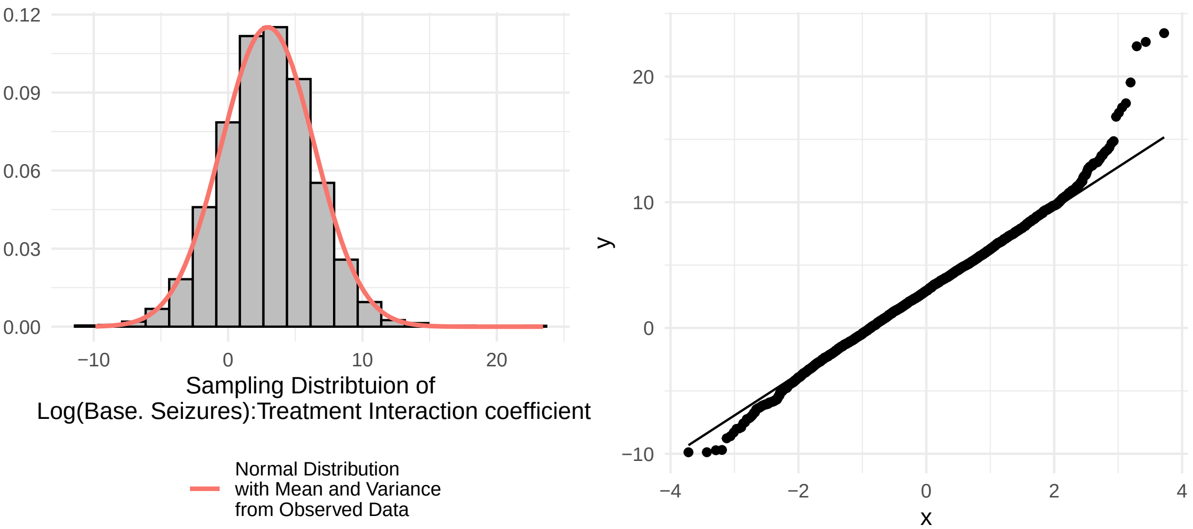

Figure 9: Sampling Distribution of respective interactive term regression coefficient

1. Normal Approximation Method

```
    2.5%     97.5%
-3.40246 10.46076
```

2. Percentile Method C.I.

```
     2.5%      97.5%
-3.818367  9.644575
```

3. Comparison with Quasipoisson

```
    2.5 %    97.5 %
-4.164202 10.921238
```

4. Comments:

- We can see that there is an issue at the upper tail, the tail is heavier than expected.
- This issue causes the discrepancy between quantile and normal approximation methods.
- Quasipoisson confidence interval is wider in each direction again.

Three-variable Interaction Term

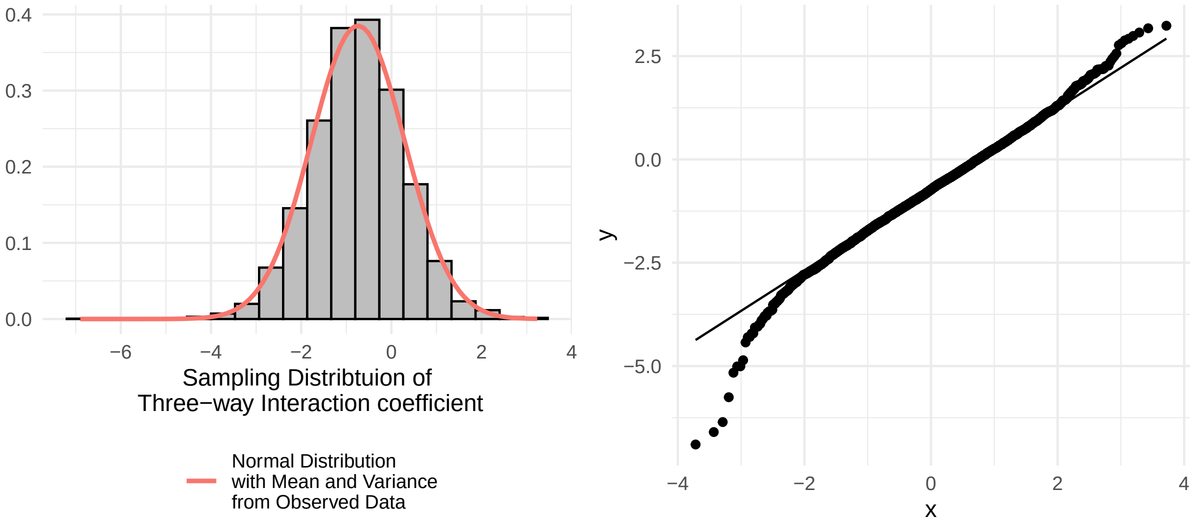

Figure 10: Sampling Distribution of a three-variable interaction term regression coefficient

1. Normal Approximation Method

```
    2.5%     97.5%
-2.94505   1.20245
```

2. Percentile Method C.I.

```
     2.5%      97.5%
-2.757267  1.243797
```

3. Comparison with Quasipoisson

```
    2.5 %     97.5 %
-3.142550  1.496276
```

4. Comments:

- We can see that there is an issue at the lower tail, the tail is heavier than expected.
- Despite the issue we have confidence interval endpoints that are quite similar. We saw that this is not always the case form previous examinations, so we will treat this occurrence as a coincidence.
- Quasipoisson confidence interval is wider in each direction again.

Problem 2

(i)

I am using code below to obtain difference curves for various values of β_1 ad β_2

```
bh <- function(S, beta1, beta2){

  1/(beta1 + beta2/S)

}
```

Figure 11 shows combinations of different parameter values and the curves that they produce. On the original scale of R and S, none of these curves seem like a good fit to the data.

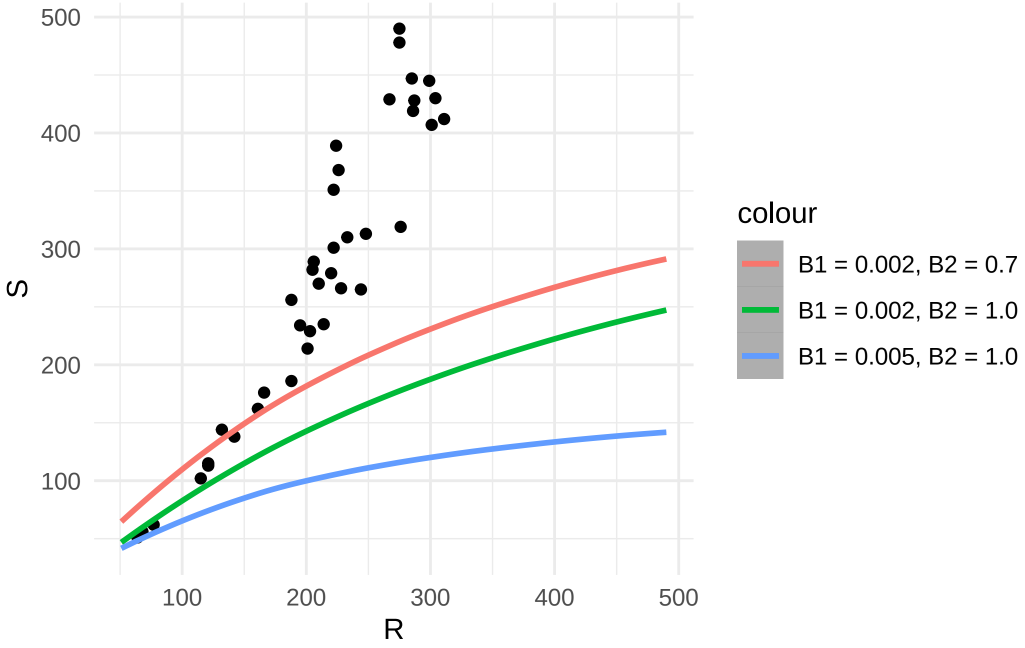

Figure 11: Various B-H curves

(ii)

Using code below we fit the mode. Summary is also provided. We will use `coef()` function later ro complete future tasks.

```
bh_lm <- lm(I(1/R) ~ I(1/S), data = fish)

summary(bh_lm)
```

```

Call:
lm(formula = I(1/R) ~ I(1/S), data = fish)

Residuals:
       Min         1Q     Median         3Q        Max 
-5.776e-04 -2.403e-04 -1.903e-05  1.755e-04  7.166e-04 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 2.013e-03  8.216e-05   24.50   <2e-16 ***
I(1/S)      6.978e-01  1.149e-02   60.72   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0003362 on 38 degrees of freedom
Multiple R-squared:  0.9898,	Adjusted R-squared:  0.9895 
F-statistic:  3687 on 1 and 38 DF,  p-value: < 2.2e-16
```

(iii)

Using a formula for the stable population that Jared provided we can write R code to compute this value

```
N = (1 - coef(bh_lm)[2])/coef(bh_lm)[1]

N
```

```
  I(1/S) 
150.0976 
```

It appears that 150 or 151 is the number of fish in each class required to maintain a stable population.

(iv)

All bootstrap work will be done with 1000 replications in the following sections.

Bootstrapped stable population estimate is provided using this function below:

```
N_hat <- function(df, est_iter) {

  res <-
    data.frame(
      i = seq(1,est_iter, 1),
      N_Stable = rep(NA, est_iter)
    )

  for(i in 1:est_iter){
    iter_bh_lm <- lm(I(1/R) ~ I(1/S), data = df[sample(rownames(df), replace = T), ])

    res$N_Stable[i] = (1 - coef(iter_bh_lm)[2])/coef(iter_bh_lm)[1]
  }

  estimate = mean(res$N_Stable)

  return(estimate)
}

N_hat(df = fish, est_iter = 1000)
```

```
[1] 150.2233
```

The answer matches exact calculation almost perfectly.

(v)

Figure 12 show the sampling distribution of the stable population N estimate.

Using this distribution and its quantiles we can obtain a percentile bootstrap confidence intervals.

The result is given below after the code chunk with a function

```
N_hat_percentile_boot <- function(df, est_iter) {
```

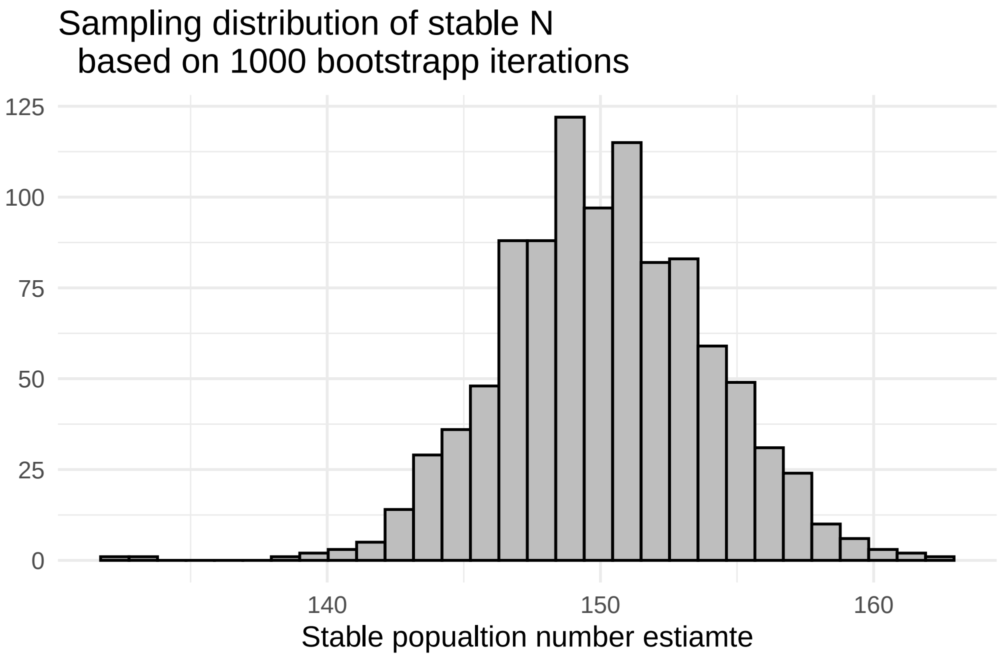

Figure 12: Sampling distribution of stable population N estiamte

```
  res <-
    data.frame(
      i = seq(1,est_iter, 1),
      N_Stable = rep(NA, est_iter)
    )

  for(i in 1:est_iter){
    iter_bh_lm <- lm(I(1/R) ~ I(1/S), data = df[sample(rownames(df), replace = T), ])

    res$N_Stable[i] = (1 - coef(iter_bh_lm)[2])/coef(iter_bh_lm)[1]
  }

  return(quantile(res$N_Stable, c(0.025, 0.975)))
}

N_hat_percentile_boot(df = fish, est_iter = 1000)
```

```
    2.5%    97.5% 
142.7234 157.4534 
```