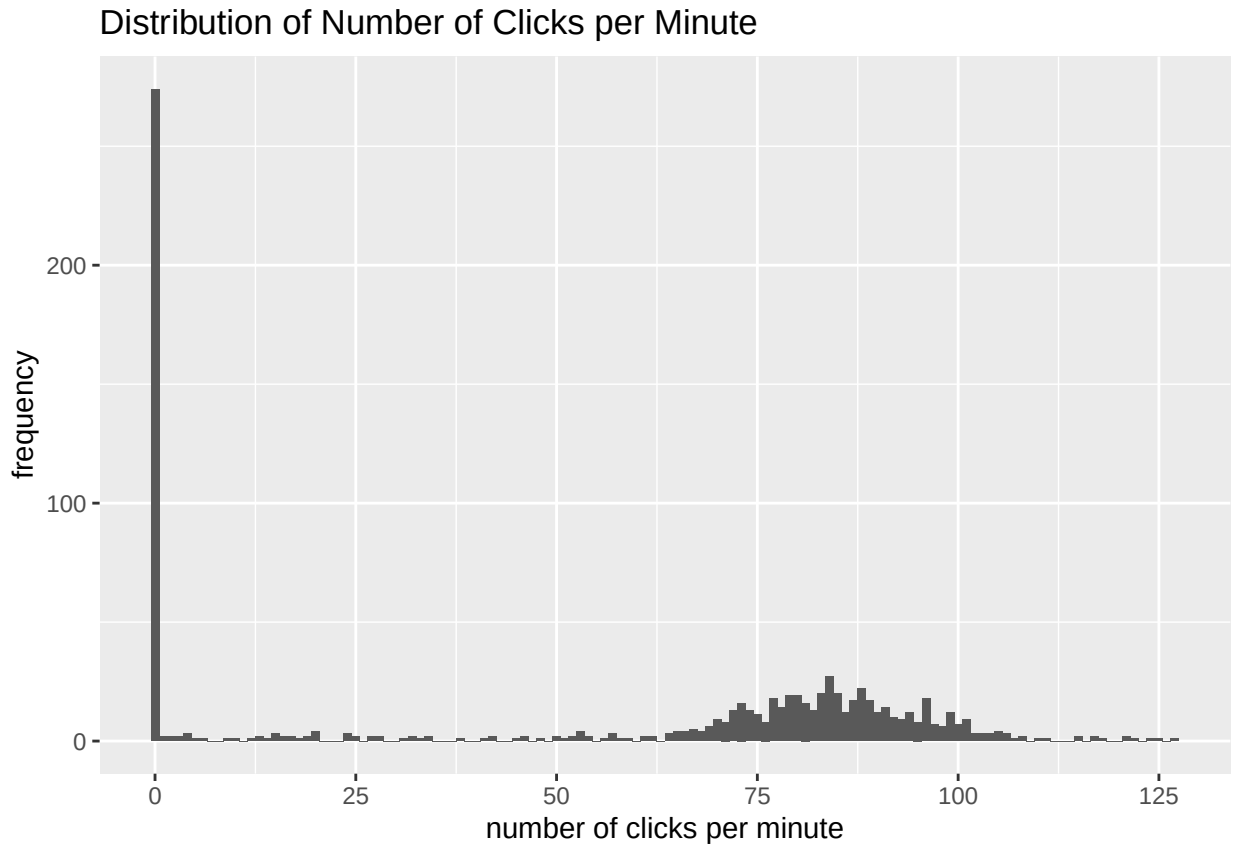
Assignment 03

2022-04-22

Question 1

Histogram of number of clicks/minute



The above histogram shows the distribution of the number of clicks per minute. You can see that the majority of the time, no clicks are occurring ie large data spike at 0. Otherwise, you can assume that the majority of whales in the study make between 70-100 clicks per minute on average. The data fits a poisson distribution as we are measuring the number of events in a given interval of time.

Calculate the average number of clicks/minute

parameter	mean
mean clicks/minute	52.79
mean clicks/second	0.88

Carrying out a bootstrap resampling procedure

statistic	mean
Bootstrap estimate of bias	-0.021
Bootstrap standard error	1.39
Bootstrap 95% CIs	50.50 - 55.07

The above confidence interval is quite different from what we see in the observed data. Firstly, the range of the data is quite small compared to the observed. I expect that the data containing the 0 (zero) values is skewing the results of our bootstrap.

Appendix of code for Question 1:

```
##histogram
mybreaks <- c(0,25,50,75,100,125)
ggplot(clicks, aes(x=n_click)) +
  geom_histogram(binwidth = 1) +
  labs(x="number of clicks per minute", y="frequency", title = "Distribution of
       Number of Clicks per Minute")+
  scale_x_continuous(breaks=mybreaks)

##calculating mean clicks/min and mean clicks/sec
mean_min <- mean(clicks$n_click)
mean_min
mean_sec <- mean_min/60
mean_sec

#bootstrapping
nreps <- 1000
## initialize empty array to hold results
bootstrap_means <- numeric(nreps)

for (i in 1:nreps) {
  bootstrap_sample <- sample(clicks$n_click, replace = TRUE)
  ##  bootstrapped mean resample
  bootstrap_means[i] <- mean(bootstrap_sample)
}
## results
results <- data.frame(bootstrap_means = bootstrap_means)
ggplot(data = results, aes(x = bootstrap_means)) +
  geom_histogram() +
  geom_vline(xintercept = as.numeric(mean)) +
  ggtitle("Sampling distribution of the mean") +
  xlab("Bootstrap means")  + ylab("") + theme_classic()

#bootstrap bias
bias <- as.numeric(mean_min) - mean(results$bootstrap_means)
bias

#bootstrap se
sd(results$bootstrap_means)

#CIs
as.numeric(mean) + c(-1,1) * qt(0.95,843)*sd(results$bootstrap_means)
```

Question 2

Manually calculate the MLE for the average number of clicks per minute (λ):

The MLE of a poisson distribution is given by

$\hat{\lambda} = \bar{x} = \frac{x_1 \ldots x_{843}}{n}$

This is equal to

$\hat{\lambda} = \frac{sum(clicks)}{843} = \frac{44499}{843}$

Therefore $\hat{\lambda} = 52.79$

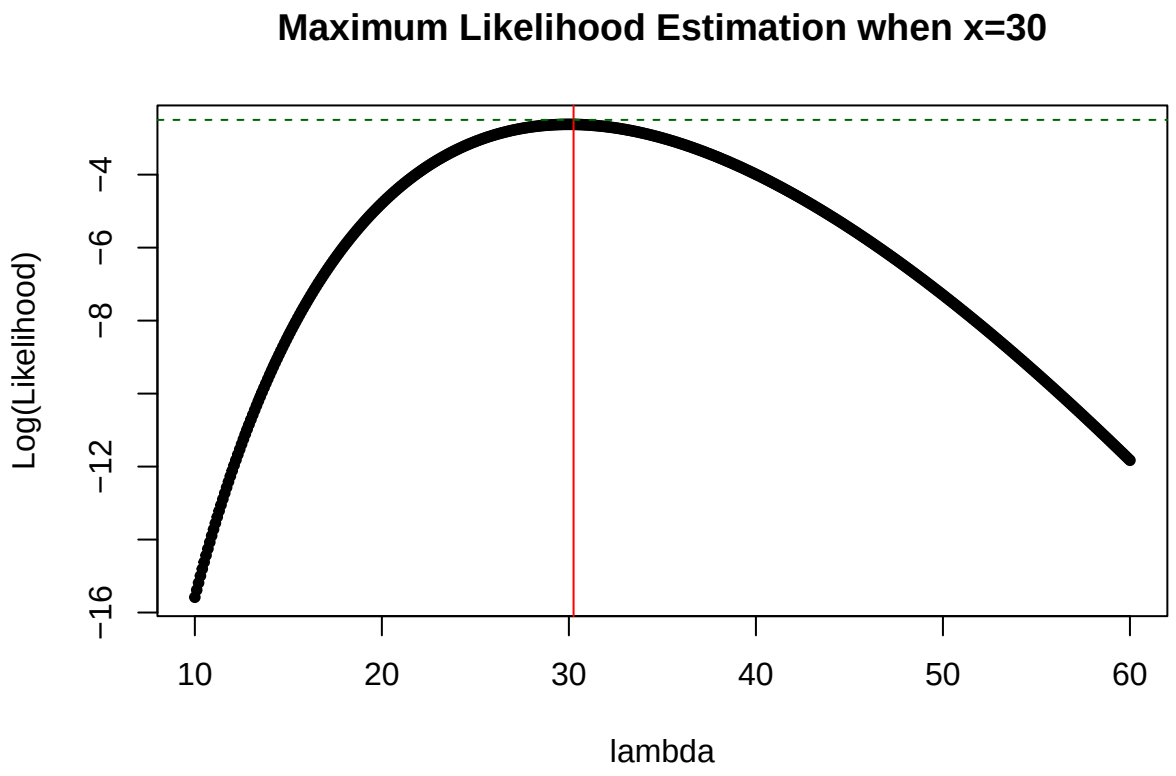
Use dpois() to form a plot

```
#use dpois
lambdaseq <- seq(10, 60, by = 0.1)
likelihoodvals <- dpois(x=30, lambdaseq, log=TRUE)

plot(lambdaseq, likelihoodvals,
     pch=20,
     xlab="lambda",
     ylab="Log(Likelihood)",
     main ="Maximum Likelihood Estimation when x=30" )
(abline(v=30.25, col="red"))
```

```
## NULL
```

```
(abline(h=-2.5, col="darkgreen", lty=2))
```



```
## NULL
```

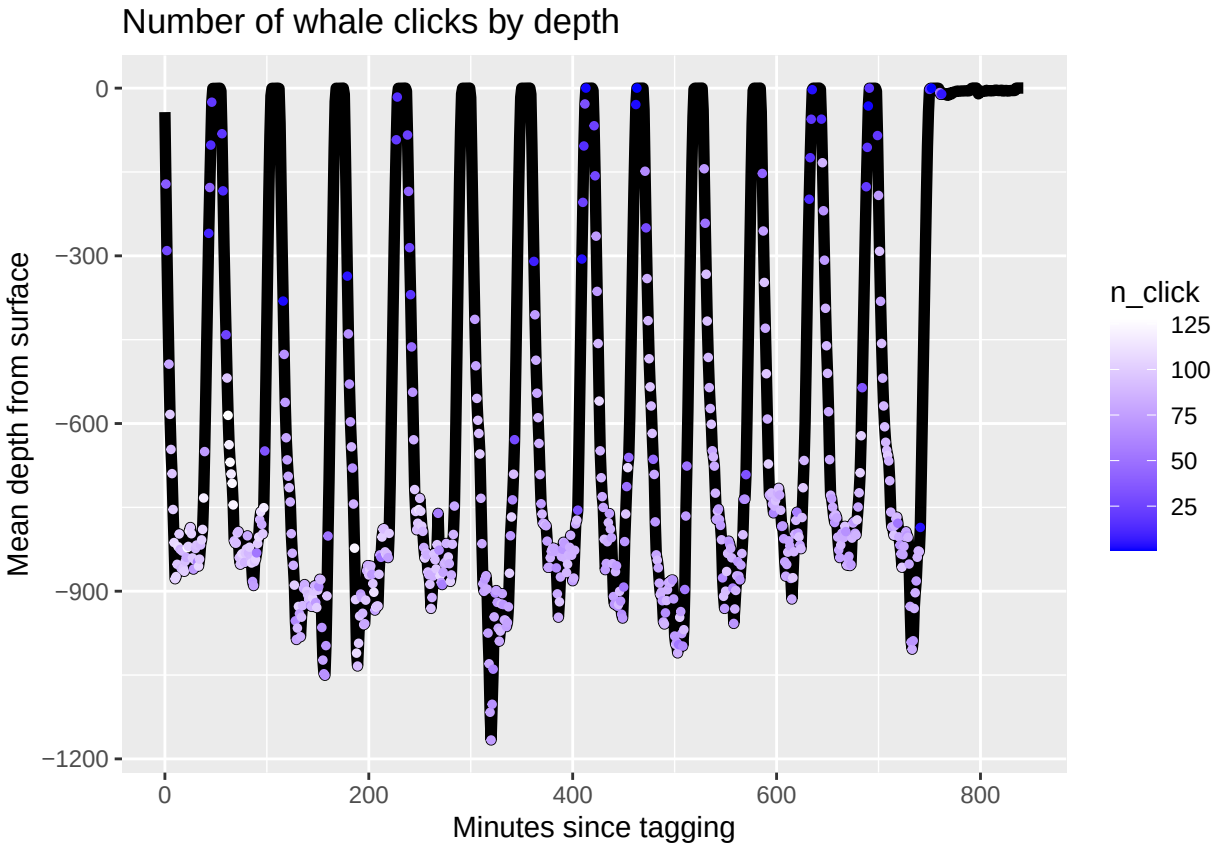
On the above plot it's evident that there is a plateau or flat point on the graph around x=30. When estimating likelihood, you look for the point at which the change in likelihood over the change in the variable (x) is equal to 0, or the rate of change is 0. This is indicated by the flat point in the top of the curve, at the intercept between the red and green lines. This MLE (~30, indicated in red) is different to what we calculated above (52.79).

My function:

```
llh <- function(lambda, obvs){
  loglikevals <- sum(dpois(lambda, obvs, log = TRUE))#creates and sums log likelihood values
  results <- data.frame(SumlogL = loglikevals, #create data frame to show sum value and parameters
                        Lambda = lambda,
                        Obesrvations = obvs)
  return(results)
}
llh(lambda, obvs)#enter values here or define earlier as objects
```

Question 3

Plot recreation



The above graph illustrates nicely the relationship between depth of the whale and the frequency or number of clicks. I have used a colour gradient to illustrate that it seems as though as depth increases, the number of clicks also increases. Additionally, very few clicks occur close to the surface level.

Fit a linear model

Linear model:

$numberofclicks = \alpha + \beta_1(meandepth) + \epsilon$

lm(n_click ~ mean_depth, data = data)

```
##                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) -3.4541879 6.97120445 -0.4954937 6.225126e-01
## mean_depth   0.1195569 0.01003696 11.9116682 6.099572e-16
```

Therefore, the fitted model is:

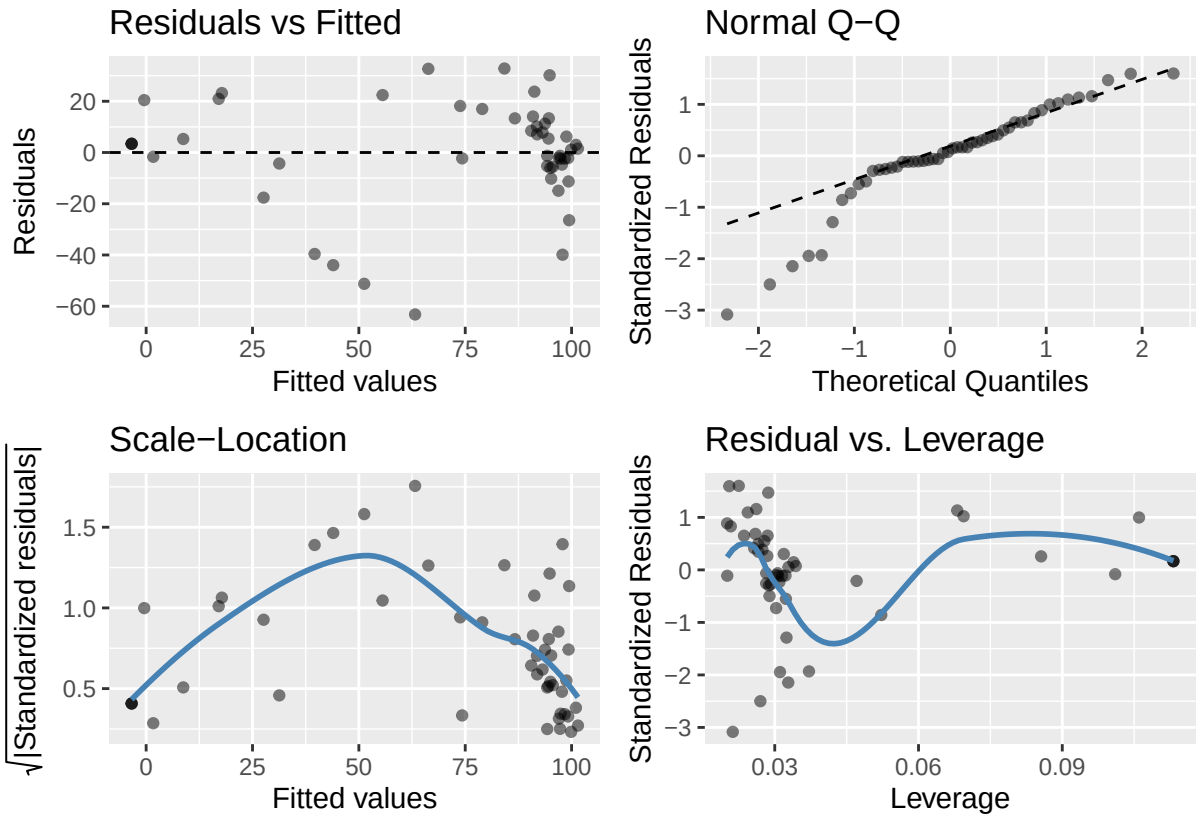
$numberofclicks = -3.45 + 0.12(meandepth) + \epsilon$

Interpretation of the coefficients:

The (Intercept) estimate (-3.45) is an estimate of the α above based on the relationship between mean depth and number of clicks. The mean_depth estimate value is the slope of mean_depth ie β. As this number is

positive (0.12), we can interpret this as mean depth increases as number of clicks increases, or vice versa. Additionally, as our p-value is small, we can assume that mean depth does influence the number of clicks.

Plot of fitted values v residuals:



The diagnostic plots are across the board suggesting that this model is not appropriate for the data.

In the residuals v fitted plot we can see there is some congregation around the zero-line indicated unequl variance ie heteroscedacity. When testing the normal distribution, we can see the data has a lot of skew on the bottom tail. This suggests that our data does not fit a normal distribution. Our scale-location plot clearly is not horizontal even roughly, which indicates that there is a lot of variance in the data, likely due to the zero-values relating to surfacing of the whales. The residuals v leverage plot shows there are a few data points that have high leverage in the data set which is not ideal.

Due to the above, I do not think the linear model just fitted is appropriate for the data.

Fit a poisson model

Formula for the model:

$log(expectednumberofclicks) = \alpha + \beta_1(meandepth)$

glm(n_click ~ mean_depth, data = data, family = "poisson")

```
##                Estimate   Std. Error  z value      Pr(>|z|)
## (Intercept) 2.307643096 0.0819382929 28.16318 1.652549e-174
## mean_depth  0.002752429 0.0001039328 26.48278 1.530665e-154
```

Therefore, the fitted model is:

$log(E(\widehat{numberofclicks}) = 2.307 + 0.0027(meandepth))$

But when we account for log of the poisson formula, the expected values are:

```
## (Intercept)  mean_depth
##   10.050708    1.002756
```

These are our expected values.

THe coefficients show the relationship between the variables, as this is positive it shows that mean depth increases as number of clicks increases, or vice versa.

Calculations of Expected Values

Expected number of clicks at the surface:

```
#clicks at the surface
clicks_0m <- 2.307 + 0.00275*(0)
exp(clicks_0m)
```

```
## [1] 10.04425
```

Expected number of clicks at 100m depth:

```
#calculate the expected clocks on the surface at an average depth of 100m
clicks_100m <- 2.307 + 0.00275*(100)
exp(clicks_100m)
```

```
## [1] 13.22356
```

Calculate the 95% CIs for predictions I've just made:

```
ci <- exp(confint(glm_whales)[1,])
ci
```

```
##     2.5 %    97.5 %
##  8.533747 11.767065
```

The above confidence interval relates to the expected values when the explanatory variable, x, is equal to 0, ie the average depth is 0. This confidence interval makes sense in that the expected value at a depth of 0m was within this range.

For the estimated λ at 100m below surface, I hae used the confidence interval estimates above and applied these to get the confidence interval:

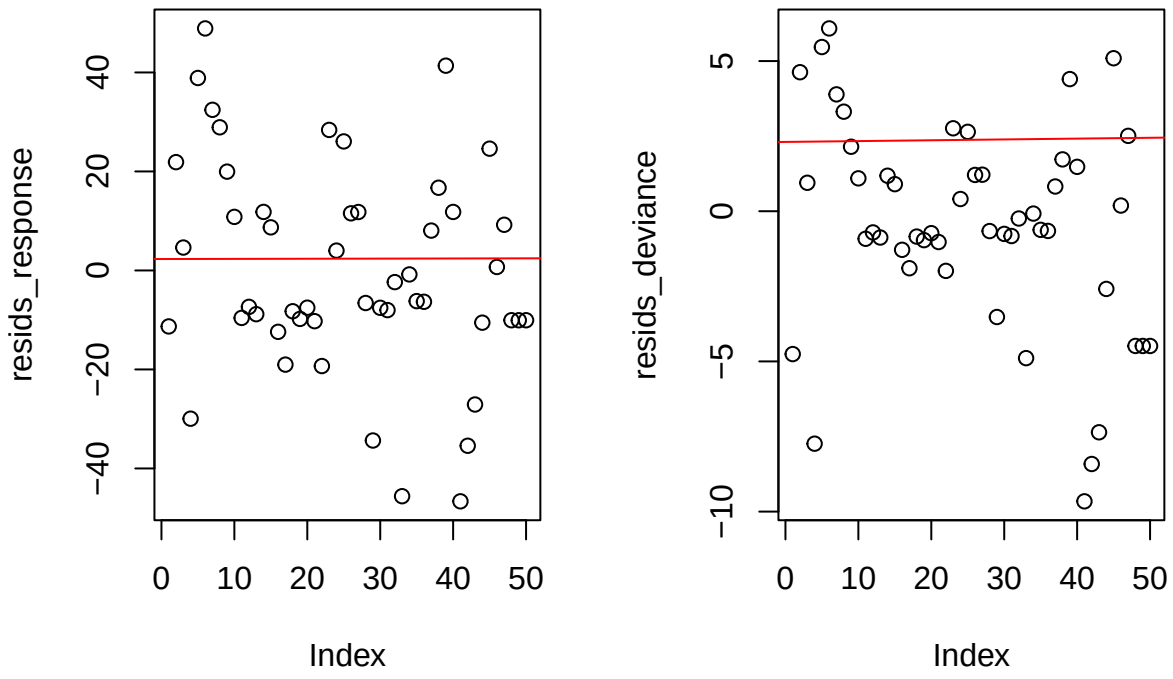
```
lower <- 13.22356-1.510503
lower
```

```
## [1] 11.71306
```

```
upper <- 13.22356+1.722815
upper
```

```
## [1] 14.94638
```

Diagnostic Plots for the poisson model:



Our assumptions for a poisson regression are as follows:

1. Variance is equal to the mean (dispersion is balanced)
2. The response variable is described by a poisson distribution
3. Independent (random) observations
4. The log of lambda is a linear function of the explanatory variable

Residual plots can be used to discover patterns or outliers in the data. We are looking for a lack of pattern in the data. Raw residuals are the difference between the observed and expected data, and model how much of the data is "missed" by our model. You can see that there is not really a pattern in the above residual plot, but that quite a lot of the data seems to stray away from the group indicating there may be a lot of outliers, or more likely showing the zero-inflation in this data set.

Pick a model that may fit better and discuss why

You should discuss why your chosen model may be more appropriate than those fitted above. You should also discuss what facets of your data you think would be best captured by your chosen modelling framework. Feel free to use illustrations in your discussion.

The data we have here is count data with inflated zeroes due to the nature of the trials. Every minute is a trial with a count, but a lot of these counts are zero due to the different depths at which these measurements are taken. The limitations of using a GML to model Poisson distributed data is that zeros are assumed to have been formed with the same process and are counted in the results in the same way. Often, these zeroes may distort our interpretation of the data.

A ZI model often assumes that the zero-data is a result of different conditions within the experiment that mean that certain groups cannot give rise to a positive or non-zero result (Feng, 2021). The ZI model also accounts for groups within the population that can give rise to a positive result but do not in the given time - this group is modeled by a normal count distribution such as poisson and can contain zeros. The hurdle model, however, assumes that two different processes occur to give rise to a result and observations must pass a "hurdle" before they can move on to the second group containing the count. Therefore, the first set of data would be a success/failure for that event, after which if there is success and the event occurs, positive count data is garnered. Hurdle model data cannot contain zeroes for this reason.

In summary, the hurdle model treats all zeroes as arising from the same process whereas the ZI model allows for two different processes that generate zeroes that should be modeled differently (Feng, 2021). In the context of the whale click data, I believe the ZI model fits better as there seems to be two distinct data-generating processes: Zeroes generated on the surface (excess zeroes)vs Zeroes and counts generated at any depth other than 0 (sampling zeroes).

Testing the models:

Zero-inflation Poisson model: zeroinfl(n_click ~ mean_depth, data = data)

```
zipm50 <- zeroinfl(n_click ~ mean_depth, data = data)
```

Vuong Test for the ZIP model with the first 50 samples compared to the Poisson GLM:

```
vuong(zipm50, glm_whales)
```

```
## Vuong Non-Nested Hypothesis Test-Statistic:
## (test-statistic is asymptotically distributed N(0,1) under the
##  null that the models are indistinguishible)
## -------------------------------------------------------------
##                Vuong z-statistic             H_A    p-value
## Raw                    3.131761 model1 > model2 0.00086881
## AIC-corrected          3.103059 model1 > model2 0.00095766
## BIC-corrected          3.075620 model1 > model2 0.00105033
```

The vuong test shows that we have significant evidence that ZIP model is a better fit than the GLM Poisson, as p=0.00087.

Fitting a hurdle model to the data:

```
hurd50 <- hurdle(n_click ~ mean_depth, data = data)
hurd50
```

```
##
## Call:
```

```
## hurdle(formula = n_click ~ mean_depth, data = data)
##
## Count model coefficients (truncated poisson with log link):
## (Intercept)   mean_depth
##    3.214254     0.001662
##
## Zero hurdle model coefficients (binomial with logit link):
## (Intercept)   mean_depth
##   -0.951536     0.005426
```

```
vuong(zipm50,hurd50)
```

```
## Vuong Non-Nested Hypothesis Test-Statistic:
## (test-statistic is asymptotically distributed N(0,1) under the
##  null that the models are indistinguishible)
## -------------------------------------------------------------
##                Vuong z-statistic             H_A p-value
## Raw               -3.738616e-06 model2 > model1     0.5
## AIC-corrected     -3.738616e-06 model2 > model1     0.5
## BIC-corrected     -3.738616e-06 model2 > model1     0.5
```

The above vuong test (p-value=0.5) shows that the null hypothesis that the models are indistinguishable is likely true and that neither test is better than the other for explaining this data. In saying that, the previous vuong test showed that either of these models explains the data better than the poisson GLM.

Based solely on how the two models interpret or view data, I believe the Zero-inflated Poisson model will be the most appropriate.

Question 4

Discuss Bayesian v Frequentist Statistics

Frequentist	Bayesian
$P(A\|B) = \frac{P(AandB)}{P(B)}$	$P(\theta\|Data) = \frac{P(Data\|\theta)P(\theta)}{P(Data)}$
-Related to the frequency of repeated events	-Related to our own certainty or uncertainty of events
-Model is fixed, data varies	-Data is fixed, model varies

The main difference between frequentist and Bayesian statistics is that Bayes' theorem and Bayesian statistics, in general, allow space for prior knowledge or belief to be a part of how we analyze the data in front of us. Conversely, frequentist statistical inferences are made based only on what is observed in the sample.

The way that Bayesian statistics allows for the inclusion of prior knowledge in statistical analysis is through a 'prior', which we can see in Bayes' theorem:

$P(\theta|Data) = \frac{P(Data|\theta)P(\theta)}{P(Data)}$

Where $P(\theta|Data)$ is a conditional or posterior probability of the event occurring, *given* the other occurring.

$P(Data|\theta)$ is also a conditional probability and can be interpreted as the likelihood.

$P(\theta), P(Data)$ are the separate probabilities of observing either event without any conditions. This is the **prior**, where our prior or "real world" knowledge comes in.

The outcome, or posterior probability, gives the probability that something (eg a hypothesis) is true, given the data we observe. This allows us to update our perceptions or belief, given new data. Additionally, this population parameter is a distribution of values reflecting uncertainty, rather than a fixed "true" value.

On the other hand, the frequentist approach aims to find the **true** value of the parameter using a fixed model and from the data only, with multiple repeated tests or experiments, ie using the long-run frequency.

The most well-known outcome of frequentist statistics is the p-value. A p-value describes the probability of observing the effect or event that you did, given the assumption that the null hypothesis (whatever that may be), is true. It does not give the probability of that event actually occurring, just how unlikely it is to occur.

Another important difference between the theories is the idea of confidence intervals. In frequentist theory, a confidence interval connotes that over an infinity of samples of the population, 95% of these contain the "true" population value. In Bayesian theory, the 'credibility interval' is more fitting, and it implies there is a 95% probability that the population value is within the limits of the interval.

References

Feng, C.X. (2021). A comparison of zero-inflated and hurdle models for modeling zero-inflated count data. *J Stat Distrib App 8, 8.* https://doi.org/10.1186/s40488-021-00121-4