Intro to statistical analyses in R

Jessica Zamborain Mason and Daniel Viana

August, 2022

Section overview

Today we are going to dive in to statistical analyses. The goal of today is to teach you how to analyze various types of datasets you are likely to encounter as a marine scientist (e.g., biomass, algal growth, presence/absence of certain species, fish count, or Likert ratings) and collected under a wide variety of study designs.

If there is something KEY I want you to learn today is that EVERY MODEL IS WRONG BUT SOME ARE USEFUL. Models take assumptions to simplify the complex reality of nature and the best skill you have to learn is to understand your model assumptions and their consequences!

Without further review, we will start. However, as always, please ask if you have any questions!

Starting R

Open RStudio. Create a new Script and save it. Remember to give it an informative name (e.g., Rintro_analyses).

Next, in your script, remove any unwanted items from your environment in order to avoid cluttering. For this course, having objects in your environment that you created on previous days is probably not a problem. However, de-cluttering is a good coding practice to make sure your script is properly following the instructions you gave it and not using past objects.

```
#clear R
rm(list=ls())
```

Also make sure you know where you are working from or set the working directory you want (e.g., where you saved the R script or where you will store the data)

```
#Get working directory
getwd()
```

```
## [1] "C:/Users/jez297/Dropbox/Harvard Postdoc/rose travel award/MADAGASCAR COURSE"
```

Alternatively,tell R to set that working directory using the "setwd()" function. In my case:

```
#set working directory
#setwd("c:/users/jzamb/Dropbox/Harvard Postdoc/rose travel award/MADAGASCAR COURSE")
setwd("C:/Users/jez297/Dropbox/Harvard Postdoc/rose travel award/MADAGASCAR COURSE")
```

Dataset for this section

As an example dataset for this section, we are going to use the dataset from Cinner et al. 2020: "Meeting fisheries, ecosystem function, and biodiversity goals in a human dominated world".

This dataset can be found in this link: https://research.jcu.edu.au/data/published/c57c53ca7177bca5a8652120d8d15928/. You likely have it already from the course coordinators. However, if you haven't, under the "Data" heading, you will find an attachment called "Cinneretal2020_multiplegoals_data.csv". Download this and store it in your working directory.

The dataset is a ".csv" file. Upload the data to R using the "read.csv()" function, making sure that when doing this, you create a data frame. We will call this data frame "reef_dat". Remember to comment your script!

```
# Upload csv data and store it as a dataframe
reef_dat<- read.csv("Cinneretal2020_multiplegoals_data.csv",header = T)
```

We created a dataframe. We told R to read the relevant .csv file from our working directory. The "header=T" is just telling R that it is "TRUE" that the file's top row is a "heading" (i.e., the different column names).

Go ahead and inspect the data. In order for you to learn the most, try to do everything yourself without looking at the answers.

```
#Display column names
colnames(reef_dat)
```

```
##  [1] "UniqueSite"                  "Site_Lat"
##  [3] "Site_Long"                   "ReefCluster"
##  [5] "Larger"                      "Biomass_above20cm"
##  [7] "Protection"                  "DepthCategory"
##  [9] "CleanHabitat"                "CensusMethod"
## [11] "Total_sampling_area"         "Regional_population_growth"
## [13] "Ocean_prod"                  "Climate_stress"
## [15] "HDI"                         "Larger_pop_size"
## [17] "Reef_fish_landings_per_km2" "MPAage3"
## [19] "NTZarea"                     "Geographic_Basin"
## [21] "Biomass...some.families"     "Local"
## [23] "Atoll"                       "Gini_lastdata"
## [25] "EPI_2018"                    "GDP"
## [27] "Control_corruption"          "Rule_of_law"
## [29] "Sedimentation"               "Annual_tourist_number"
## [31] "Locality"                    "gravtot5001"
## [33] "gravtot5002"                 "gravtot5003"
## [35] "Trait_diversity"             "Scraping_potential"
## [37] "MPAage"
```

```
#Display subset of data
head(reef_dat)
```

```
##   UniqueSite  Site_Lat Site_Long ReefCluster    Larger Biomass_above20cm
## 1          4  1.501789  125.2616       RC_22 Indonesia        373.567854
## 2          5  1.500622  125.2429       RC_22 Indonesia        315.967018
## 3          6  1.585250  118.5872       RC_32 Indonesia        525.134694
## 4          7 -8.367940  116.0978        RC_1 Indonesia          4.832758
```

```
## 5          9  1.505397  125.2491       RC_22 Indonesia        119.408986
## 6         10  1.519163  125.2407       RC_22 Indonesia          5.107125
##   Protection DepthCategory CleanHabitat           CensusMethod
## 1     Fished         4-10m        Slope Standard belt transect
## 2     Fished         4-10m        Slope Standard belt transect
## 3 Restricted          0-4m        Crest Standard belt transect
## 4     Fished          0-4m        Crest Standard belt transect
## 5     Fished         4-10m        Slope Standard belt transect
## 6     Fished         4-10m        Slope Standard belt transect
##   Total_sampling_area Regional_population_growth Ocean_prod Climate_stress
## 1                 750                   8.642913      0.193      0.8474117
## 2                 750                   8.642913      0.193      0.8474117
## 3                 750                  21.216517      0.349      0.7890314
## 4                 750                  11.401162      0.284      0.8274844
## 5                 750                   8.642913      0.193      0.8474117
## 6                 750                   8.642913      0.193      0.8474117
##     HDI Larger_pop_size Reef_fish_landings_per_km2 MPAage3 NTZarea
## 1 0.662       240676485                    6.67873       0       0
## 2 0.662       240676485                    6.67873       0       0
## 3 0.662       240676485                    6.67873       0       0
## 4 0.662       240676485                    6.67873       0       0
## 5 0.662       240676485                    6.67873       0       0
## 6 0.662       240676485                    6.67873       0       0
##       Geographic_Basin Biomass...some.families     Local Atoll Gini_lastdata
## 1 Central Indo-Pacific               619.62614 Indonesia     0          39.5
## 2 Central Indo-Pacific               394.05620 Indonesia     0          39.5
## 3 Central Indo-Pacific               737.29001 Indonesia     0          39.5
## 4 Central Indo-Pacific                20.37535 Indonesia     0          39.5
## 5 Central Indo-Pacific               182.09886 Indonesia     0          39.5
## 6 Central Indo-Pacific               326.14908 Indonesia     0          39.5
##   EPI_2018  GDP Control_corruption Rule_of_law Sedimentation
## 1    46.92 4923              -0.74       -0.64     0.2985979
## 2    46.92 4923              -0.74       -0.64     0.2985979
## 3    46.92 4923              -0.74       -0.64     0.5482134
## 4    46.92 4923              -0.74       -0.64     0.3213682
## 5    46.92 4923              -0.74       -0.64     0.2985979
## 6    46.92 4923              -0.74       -0.64     0.2985979
##   Annual_tourist_number  Locality gravtot5001 gravtot5002  gravtot5003
## 1                  3788 Indonesia   17331.379  127.669722   2.29520457
## 2                  3788 Indonesia   14817.132   76.749220   0.55229701
## 3                  3788      <NA>    8489.429    8.758425   0.01112817
## 4                  3788 Indonesia  122299.990 1049.194849 114.38935625
## 5                  3788 Indonesia   17490.595  133.262453   2.76630042
## 6                  3788 Indonesia   16667.977  109.046147   1.22218168
##   Trait_diversity Scraping_potential MPAage
## 1        2.688359          115.13790      0
## 2        3.054261           11.03998      0
## 3        3.403984          864.69787      0
## 4        2.590875           12.49623      0
## 5        3.458912           54.74287      0
## 6        1.699385           53.78552      0
```

```
# Display structure of the data
str(reef_dat)
```

```
## 'data.frame':    1798 obs. of  37 variables:
##  $ UniqueSite               : int  4 5 6 7 9 10 11 12 13 21 ...
##  $ Site_Lat                 : num  1.5 1.5 1.59 -8.37 1.51 ...
##  $ Site_Long                : num  125 125 119 116 125 ...
##  $ ReefCluster              : chr  "RC_22" "RC_22" "RC_32" "RC_1" ...
##  $ Larger                   : chr  "Indonesia" "Indonesia" "Indonesia" "Indonesia" ...
##  $ Biomass_above20cm        : num  373.57 315.97 525.13 4.83 119.41 ...
##  $ Protection               : chr  "Fished" "Fished" "Restricted" "Fished" ...
##  $ DepthCategory            : chr  "4-10m" "4-10m" "0-4m" "0-4m" ...
##  $ CleanHabitat             : chr  "Slope" "Slope" "Crest" "Crest" ...
##  $ CensusMethod             : chr  "Standard belt transect" "Standard belt transect" "Standard belt
##  $ Total_sampling_area      : num  750 750 750 750 750 750 750 750 750 750 ...
##  $ Regional_population_growth: num  8.64 8.64 21.22 11.4 8.64 ...
##  $ Ocean_prod               : num  0.193 0.193 0.349 0.284 0.193 0.193 0.192 0.193 0.193 0.324 ...
##  $ Climate_stress           : num  0.847 0.847 0.789 0.827 0.847 ...
##  $ HDI                      : num  0.662 0.662 0.662 0.662 0.662 0.662 0.662 0.662 0.662 0.662 ...
##  $ Larger_pop_size          : int  240676485 240676485 240676485 240676485 240676485 240676485 2406
##  $ Reef_fish_landings_per_km2: num  6.68 6.68 6.68 6.68 6.68 ...
##  $ MPAage3                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ NTZarea                  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Geographic_Basin         : chr  "Central Indo-Pacific" "Central Indo-Pacific" "Central Indo-Paci
##  $ Biomass...some.families  : num  619.6 394.1 737.3 20.4 182.1 ...
##  $ Local                    : chr  "Indonesia" "Indonesia" "Indonesia" "Indonesia" ...
##  $ Atoll                    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gini_lastdata            : num  39.5 39.5 39.5 39.5 39.5 39.5 39.5 39.5 39.5 39.5 ...
##  $ EPI_2018                 : num  46.9 46.9 46.9 46.9 46.9 ...
##  $ GDP                      : int  4923 4923 4923 4923 4923 4923 4923 4923 4923 4923 ...
##  $ Control_corruption       : num  -0.74 -0.74 -0.74 -0.74 -0.74 -0.74 -0.74 -0.74 -0.74 -0.74 ...
##  $ Rule_of_law              : num  -0.64 -0.64 -0.64 -0.64 -0.64 -0.64 -0.64 -0.64 -0.64 -0.64 ...
##  $ Sedimentation            : num  0.299 0.299 0.548 0.321 0.299 ...
##  $ Annual_tourist_number    : num  3788 3788 3788 3788 3788 ...
##  $ Locality                 : chr  "Indonesia" "Indonesia" NA "Indonesia" ...
##  $ gravtot5001              : num  17331 14817 8489 122300 17491 ...
##  $ gravtot5002              : num  127.67 76.75 8.76 1049.19 133.26 ...
##  $ gravtot5003              : num  2.2952 0.5523 0.0111 114.3894 2.7663 ...
##  $ Trait_diversity          : num  2.69 3.05 3.4 2.59 3.46 ...
##  $ Scraping_potential       : num  115.1 11 864.7 12.5 54.7 ...
##  $ MPAage                   : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
#summarize the data
summary(reef_dat)
```

```
##    UniqueSite      Site_Lat         Site_Long       ReefCluster
##  Min.   :   4   Min.   :-23.420   Min.   :-179.91   Length:1798
##  1st Qu.:2955   1st Qu.:-18.401   1st Qu.:-155.60   Class :character
##  Median :5570   Median : -5.768   Median :  51.19   Mode  :character
##  Mean   :5619   Mean   : -3.299   Mean   :  12.57
##  3rd Qu.:8482   3rd Qu.:  6.435   3rd Qu.: 145.84
##  Max.   :9560   Max.   : 23.430   Max.   : 179.73
##
##     Larger          Biomass_above20cm   Protection        DepthCategory
##  Length:1798        Min.   :    0.00   Length:1798        Length:1798
##  Class :character   1st Qu.:   75.94   Class :character   Class :character
##  Mode  :character   Median :  257.36   Mode  :character   Mode  :character
```

```
##                    Mean   :  721.78
##                    3rd Qu.:  692.44
##                    Max.   :22742.91
##
##  CleanHabitat       CensusMethod       Total_sampling_area
##  Length:1798        Length:1798        Min.   :  50.0
##  Class :character   Class :character   1st Qu.: 400.0
##  Mode  :character   Mode  :character   Median : 500.0
##                                        Mean   : 611.8
##                                        3rd Qu.: 750.0
##                                        Max.   :2500.0
##
##  Regional_population_growth   Ocean_prod     Climate_stress        HDI
##  Min.   :-23.459            Min.   :0.0320   Min.   :0.2120   Min.   :0.3970
##  1st Qu.:  6.106            1st Qu.:0.0810   1st Qu.:0.4086   1st Qu.:0.6620
##  Median : 13.333            Median :0.1495   Median :0.5807   Median :0.8820
##  Mean   : 13.325            Mean   :0.1596   Mean   :0.5606   Mean   :0.7886
##  3rd Qu.: 19.705            3rd Qu.:0.2030   3rd Qu.:0.7422   3rd Qu.:0.9100
##  Max.   : 60.909            Max.   :3.2870   Max.   :0.9371   Max.   :0.9270
##
##  Larger_pop_size     Reef_fish_landings_per_km2    MPAage3
##  Min.   :        0   Min.   :  0.00085          Min.   : 0.0000
##  1st Qu.:    97743   1st Qu.:  0.85702          1st Qu.: 0.0000
##  Median :   325694   Median :  1.80350          Median : 0.0000
##  Mean   : 34223407   Mean   :  6.40823          Mean   : 0.9318
##  3rd Qu.: 22065300   3rd Qu.:  5.16710          3rd Qu.: 0.0000
##  Max.   :240676485   Max.   :132.86369          Max.   :43.0000
##
##     NTZarea       Geographic_Basin   Biomass...some.families    Local
##  Min.   :     0   Length:1798        Min.   :    1.005       Length:1798
##  1st Qu.:     0   Class :character   1st Qu.:  175.072       Class :character
##  Median :     0   Mode  :character   Median :  383.307       Mode  :character
##  Mean   : 12824                      Mean   :  794.783
##  3rd Qu.:     0                      3rd Qu.:  837.488
##  Max.   :640000                      Max.   :21276.611
##
##      Atoll        Gini_lastdata      EPI_2018          GDP
##  Min.   :0.0000   Min.   :28.60   Min.   :33.73   Min.   :  945
##  1st Qu.:0.0000   1st Qu.:34.25   1st Qu.:52.14   1st Qu.: 6241
##  Median :0.0000   Median :39.50   Median :71.19   Median :22000
##  Mean   :0.2575   Mean   :38.64   Mean   :66.02   Mean   :23303
##  3rd Qu.:1.0000   3rd Qu.:41.00   3rd Qu.:74.12   3rd Qu.:37700
##  Max.   :1.0000   Max.   :53.30   Max.   :83.95   Max.   :51704
##                   NA's   :28      NA's   :17
##  Control_corruption  Rule_of_law      Sedimentation     Annual_tourist_number
##  Min.   :-1.2500    Min.   :-1.6400   Min.   :0.00000   Min.   :    0
##  1st Qu.:-0.4450    1st Qu.:-0.5200   1st Qu.:0.07207   1st Qu.:   51
##  Median : 1.2300    Median : 1.5400   Median :0.24169   Median :  143
##  Mean   : 0.5738    Mean   : 0.7158   Mean   :0.22438   Mean   : 1699
##  3rd Qu.: 1.4200    3rd Qu.: 1.6000   3rd Qu.:0.32706   3rd Qu.: 2596
##  Max.   : 2.0600    Max.   : 1.7700   Max.   :0.77116   Max.   :11640
##
##    Locality          gravtot5001      gravtot5002         gravtot5003
##  Length:1798        Min.   :     0   Min.   :    0.000   Min.   :   0.000
```

```
##  Class :character   1st Qu.:   331   1st Qu.:    1.210   1st Qu.:   0.009
##  Mode  :character   Median :  1332   Median :    9.679   Median :   0.164
##                     Mean   : 13268   Mean   :  163.001   Mean   :  51.355
##                     3rd Qu.:  8015   3rd Qu.:   78.093   3rd Qu.:   3.869
##                     Max.   :204108   Max.   :10722.689   Max.   :9738.842
##
##  Trait_diversity Scraping_potential     MPAage
##  Min.   :1.000   Min.   :   0.00    Min.   : 0.0000
##  1st Qu.:1.905   1st Qu.:  10.72    1st Qu.: 0.0000
##  Median :2.440   Median :  82.67    Median : 0.0000
##  Mean   :2.588   Mean   : 159.43    Mean   : 0.9318
##  3rd Qu.:3.123   3rd Qu.: 213.44    3rd Qu.: 0.0000
##  Max.   :5.413   Max.   :5197.42    Max.   :43.0000
##  NA's   :136     NA's   :136
```

QUESTION: What format does the data have? (wide or long?)

These data are in wide format, each row is a unique reef site, and each column is a variable associated to that site. It is important that you always inspect your data first. We are going to leave it as a wide format for our explorations today.

There is lots of information there.

Basically, the dataset consists of 1798 reef sites spanned around the globe. For each reef site there is (i) sampling and environmental information like the country or depth of the survey, (ii) socio-economic information like population size or human impact (gravtot5002), and (iii) four key ecological metrics important for reef ecosystems: biomass of reef fish, biomass of reef fish >20 cm, trait diversity, and parrotfish scraping potential ("Biomass. . . some.families", "Biomass_above20cm","Trait_diversity" and "Scraping_potential", respectively). The summary function shows the ranges of the variables that are numeric. However, there are also variables that are categorical like reef habitat ("CleanHabitat").

Summary statistics

Next we are going to compute various statistics in R that might be of interest to you for your own work.

Go ahead and calculate the mean, and the median of "Biomass. . . some.families"

```
#calculate mean
mean(reef_dat$Biomass...some.families)
```

```
## [1] 794.7828
```

```
#calculate median
median(reef_dat$Biomass...some.families)
```

```
## [1] 383.3072
```

The mean is a measure of the central location of the data values (in this case the average biomass of reef fish in the data base is ~ 795 kg/ha). The median is the middle value when the data is ordered from least to greatest (in this case the median biomass of reef fish in the database is ~383 kg/ha).

Calculate the range, the variance and the standard deviation.

```
#calculate range
max(reef_dat$Biomass...some.families)-min(reef_dat$Biomass...some.families)
```

```
## [1] 21275.61
```

```
#calculate standard deviation
sd(reef_dat$Biomass...some.families)
```

```
## [1] 1386.002
```

```
#calculate variance
sd(reef_dat$Biomass...some.families)^2
```

```
## [1] 1921001
```

The range is the difference of the variable's largest and smallest data values, which is a measure of the spread of biomass in this dataset. The variance is a measure of how the data values are dispersed around the mean, a measure of the amount of variation (how values deviate from the mean). The standard deviation is the squared root of the variance.

There are other statistics you can calculate for a given variable. For example, qunatiles. Quantiles are points in a distribution that relate to the rank order of values in that distribution. Percentiles are descriptions of quantiles relative to 100; so the 75th percentile (upper quartile) is 75% or three quarters of the way up an ascending list of sorted values of a sample. The 25th percentile (lower quartile) is one quarter of the way up this rank order.

```
#calculate quantalis
quantile(reef_dat$Biomass...some.families)
```

```
##           0%          25%          50%          75%         100%
##     1.005224   175.072428   383.307173   837.487936 21276.611280
```

Here we calculated quartiles.The first quartile, or lower quartile, is the value that cuts off the first 25% of the data when it is sorted in ascending order. The second quartile, or median, is the value that cuts off the first 50%. The third quartile, or upper quartile, is the value that cuts off the first 75%.

QUESTION: Previously we calculated the mean and the median. Can you think of a reason why the median and mean are so different in this example?

The median and mean differ substantially when the distribution is skewed (i.e., not symmetrical). We will dig more into this later, but when a distribution is skewed, the median does a better job of describing the center of the distribution than the mean.

To know if our variable of interest is skewed we can calculate the skewness. In R we can do this with the "e1071" package. Load that library (install it if you do not already have it) and calculate the skewness of biomass:

```
#load library
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 4.2.2
```

```r
#calculate skewness
skewness(reef_dat$Biomass...some.families)
```

```
## [1] 5.995476
```

The skewness is a measure of symmetry. Negative skewness would indicate that the mean biomass is less than the median, and the biomass distribution is left-skewed. Positive skewness would indicate that the mean biomass is larger than the median, and the biomass distribution is right-skewed.

QUESTION: Is our biomass variable symmetrical? if not, is the variable left or right-skewed?

If you answered that our Biomass variable is not symmetrical and right-skewed you are correct! well-done! Knowing how your data is distributed is very important for your analyses.

Distributions: knowing the data-generation process

A probability distribution describes how the values of a random variable are distributed.

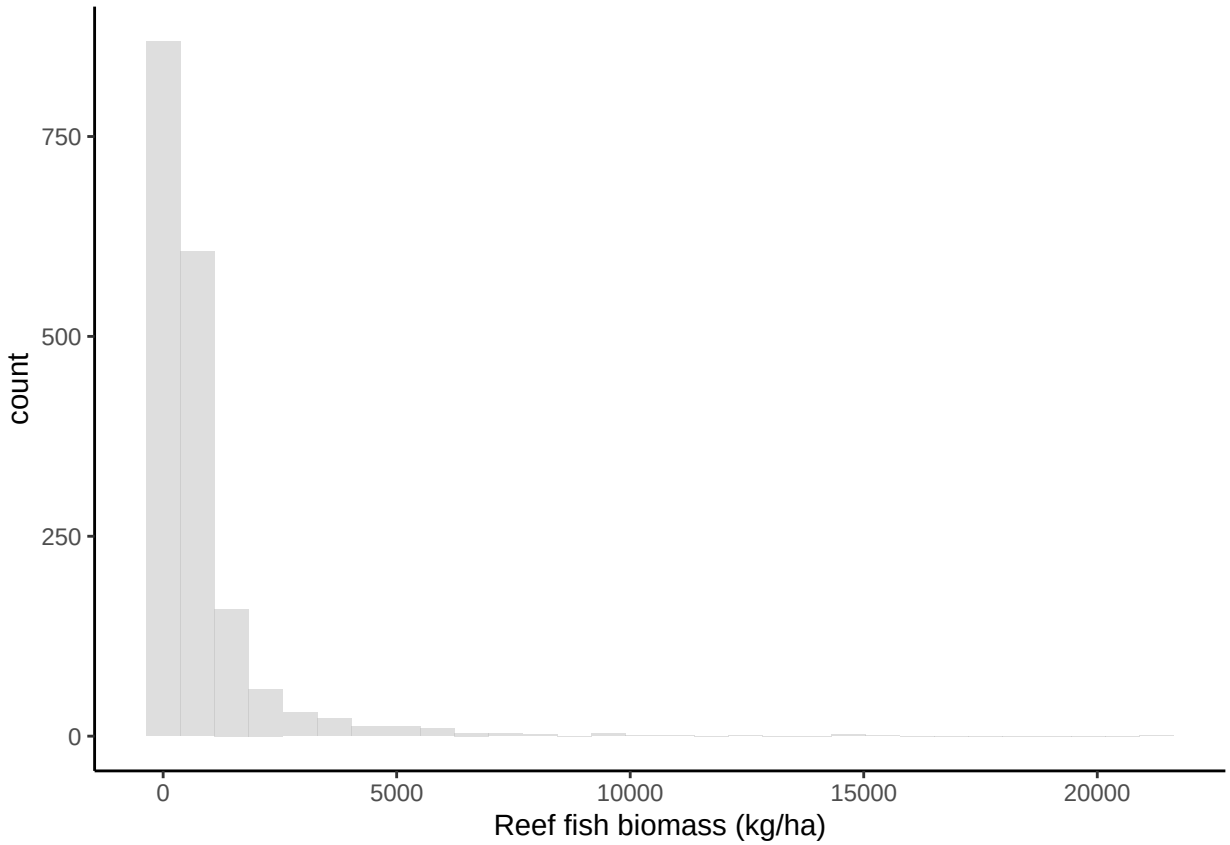
It is always good practice to visualize the distribution of your variable of interest. Go ahead and create an histogram of biomass using the ggplot package (also within the tidyverse package)

```r
#load library
#library(tidyverse)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```r
#plot distribution
ggplot(reef_dat,aes(x=Biomass...some.families))+
  geom_histogram(fill="grey",alpha=0.5)+
  theme_classic()+xlab ("Reef fish biomass (kg/ha)")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



As you can see, you can confirm that the distribution of biomass is right-skewed, which is telling you that the majority of sites have biomass values at the lower range whereas very few sites have high biomass values.

There are different well-known probability distributions. The data that we collect often resembles one of these distributions (data generation process). Knowing your variable's distribution is very important for your statistical analyses. Some common distributions you will encounter in marine science are the normal and lognormal distributions (e.g., for continuous data like biomass of reef fish), the binomial distribution (e.g., for binary outcomes like the presence/absence of a given invertebrate group), and the poisson and negative binomial distributions (e.g.,for count data like number of coral organisms).

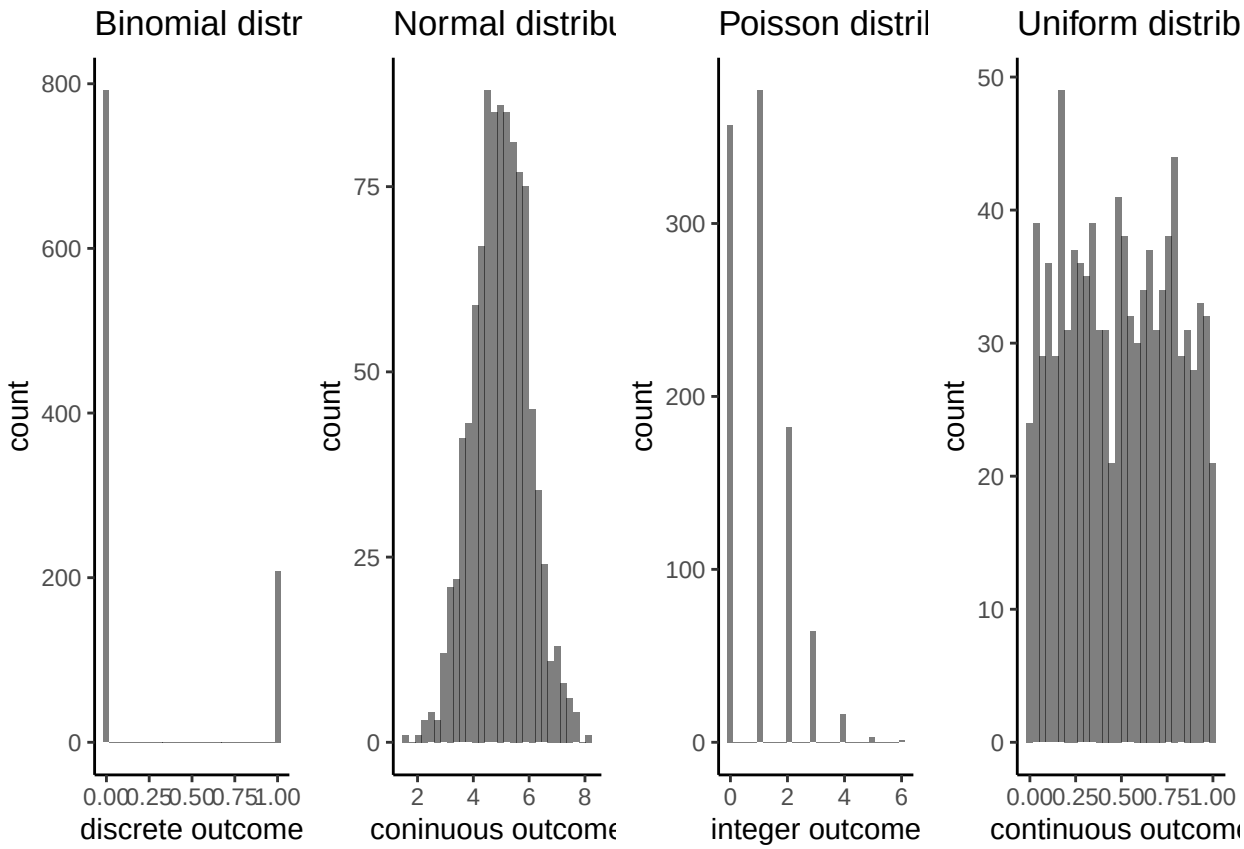
Let's go ahead and visualize some well-known distributions in R. To do this we will simulate data. Simulation, creating new data with a given set of specified parameters, is very useful in statistical analyses, for example to make sure your models are doing what they are supposed to. We will not go into a lot of simulation today, but I thought I would flag in case some of you would like to look into that topic further at another stage.

```
#generate data from different distributions with a sample size of 1000
#binomial distribution: sample size=1000, number of trials, and probability of success (e.g., presence)
binom_dat<-rbinom(1000,size=1,prob=0.2)
#normal distribution:sample size=1000, mean=5 and standard deviation=1
norm_dat<-rnorm(1000,5,1)
#poisson distribution:sample size=1000, lambda=1
pois_dat<-rpois (1000,1)
#uniform distribution:sample size=1000, min=0, max=1
unif_dat<-runif (1000,0,1)

#load ggpubr r to plot distributions together
library(ggpubr)
```

```
#plot distributions together
a<-ggplot()+
  geom_histogram(aes(x=binom_dat),fill="black",alpha=0.5)+
  theme_classic()+
  xlab ("discrete outcome")+ggtitle("Binomial distribution")
b<-ggplot()+geom_histogram(aes(x=norm_dat),fill="black",alpha=0.5)+
  theme_classic()+
  xlab ("coninuous outcome")+
  ggtitle("Normal distribution")
c<-ggplot()+geom_histogram(aes(x=pois_dat),fill="black",alpha=0.5)+
  theme_classic()+
  xlab ("integer outcome")+
  ggtitle("Poisson distribution")
d<-ggplot()+
  geom_histogram(aes(x=unif_dat),fill="black",alpha=0.5)+
  theme_classic()+
  xlab ("continuous outcome")+
  ggtitle("Uniform distribution")
windows()
ggarrange(a,b,c,d,nrow=1,ncol=4)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Ok, let me tell you what we did here. We created random new data from different probability distributions specifying the sample size and distribution parameters. Then we generated different plots for each distribution and combined them using the ggarrange() function. Use the help() or ? functions to understand what those functions do if you need to.

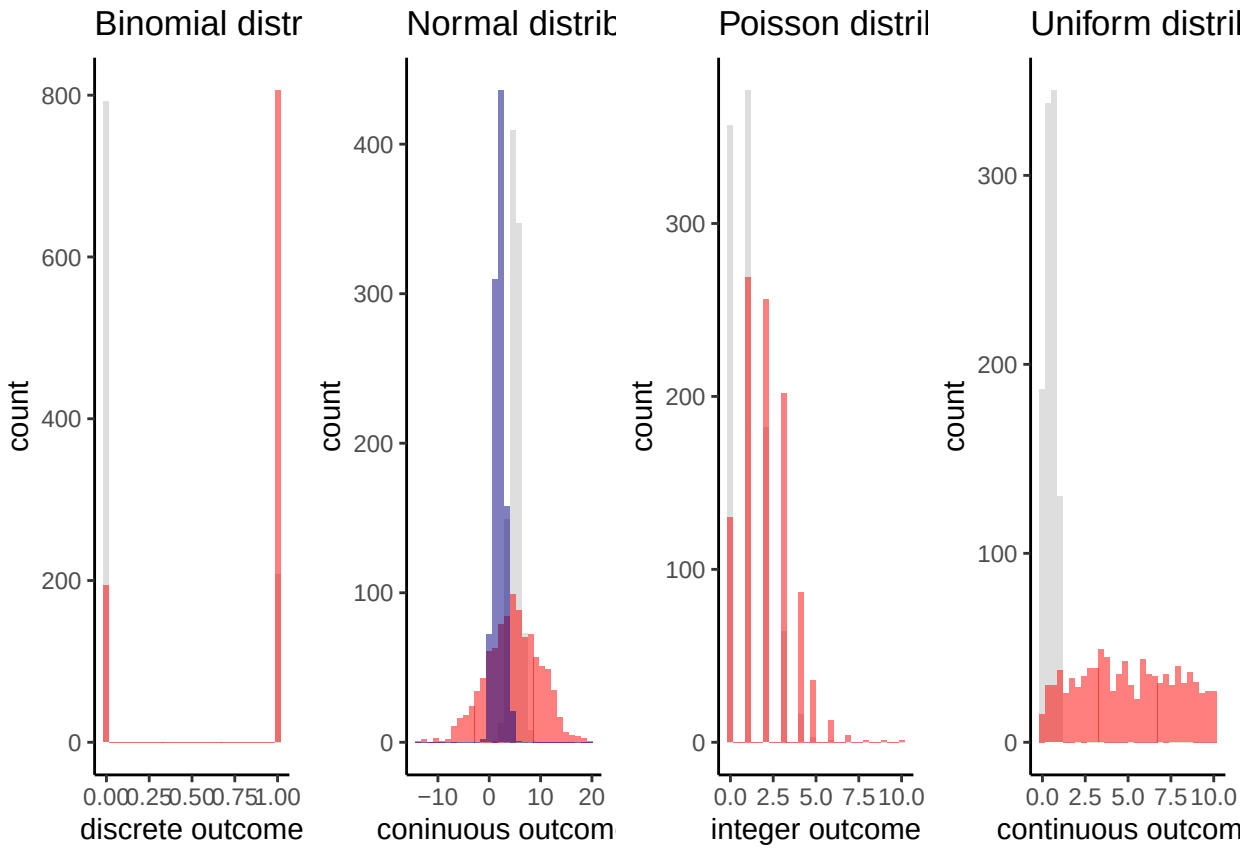
You can play around with the parameters you that you generated your data from and try to understand how changing parameters changes the distributions:

For example, for the binomial distribution, change the probability of success and for the normal distribution change the mean and/or standard deviation.

Here are some examples:

```
#change probability of success
a<-ggplot()+geom_histogram(aes(x=binom_dat),fill="grey",alpha=0.5)+theme_classic()+xlab ("discrete outc
  geom_histogram(aes(x=rbinom(1000,size=1,prob=0.8)),fill="red",alpha=0.5)
#change mean or standard deviation
b<-ggplot()+geom_histogram(aes(x=norm_dat),fill="grey",alpha=0.5)+theme_classic()+xlab ("coninuous outc
  geom_histogram(aes(x=rnorm(1000,5,5)),fill="red",alpha=0.5)+   geom_histogram(aes(x=rnorm(1000,2,1)),
#change lambda
c<-ggplot()+geom_histogram(aes(x=pois_dat),fill="grey",alpha=0.5)+theme_classic()+xlab ("integer outcom
  geom_histogram(aes(x=rpois(1000,2)),fill="red",alpha=0.5)
#change range
d<-ggplot()+geom_histogram(aes(x=unif_dat),fill="grey",alpha=0.5)+theme_classic()+xlab ("continuous out
  geom_histogram(aes(x=runif(1000,0,10)),fill="red",alpha=0.5)
#plot distributions together
ggarrange(a,b,c,d,nrow=1,ncol=4)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Many times our data do not directly follow one of these common distributions. However, sometimes, for some analyses, it may be useful to transform our response variable so it does (always making sure you understand what has been done, making sure all variables have been transformed successfully (e.g., you did not introduce NAs or Infs with your transformation), and account for the transformation in your interpretations).

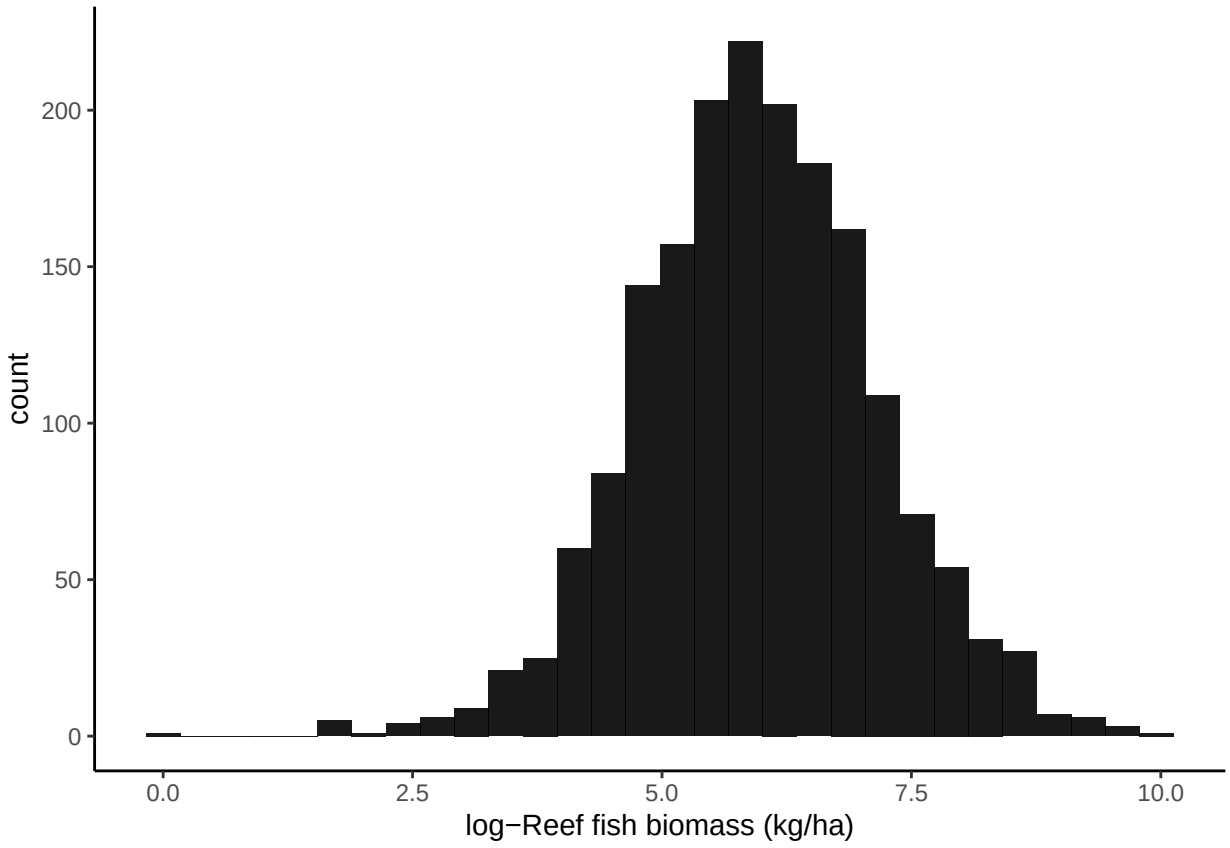
This is the case with our biomass variable. It is a continuous variable that does not have zeroes. I mentioned before it seems to follow a log normal distribution, which means that the log of our biomass variable may follow a normal distribution. Go ahead, check that the biomass variable does not have any zeroes, create a new variable called "log_biomass" that is the log-transformed version of the biomass variable and plot the distribution of log biomass.

```
#create a new variable with transformed biomass
summary(reef_dat$Biomass...some.families)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##     1.005   175.072   383.307   794.783   837.488 21276.611
```

```
reef_dat$log_biomass<-log(reef_dat$Biomass...some.families)
#plot distribution
ggplot(reef_dat,aes(x=log_biomass))+geom_histogram(fill="black",alpha=0.9)+theme_classic()+xlab ("log-R
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

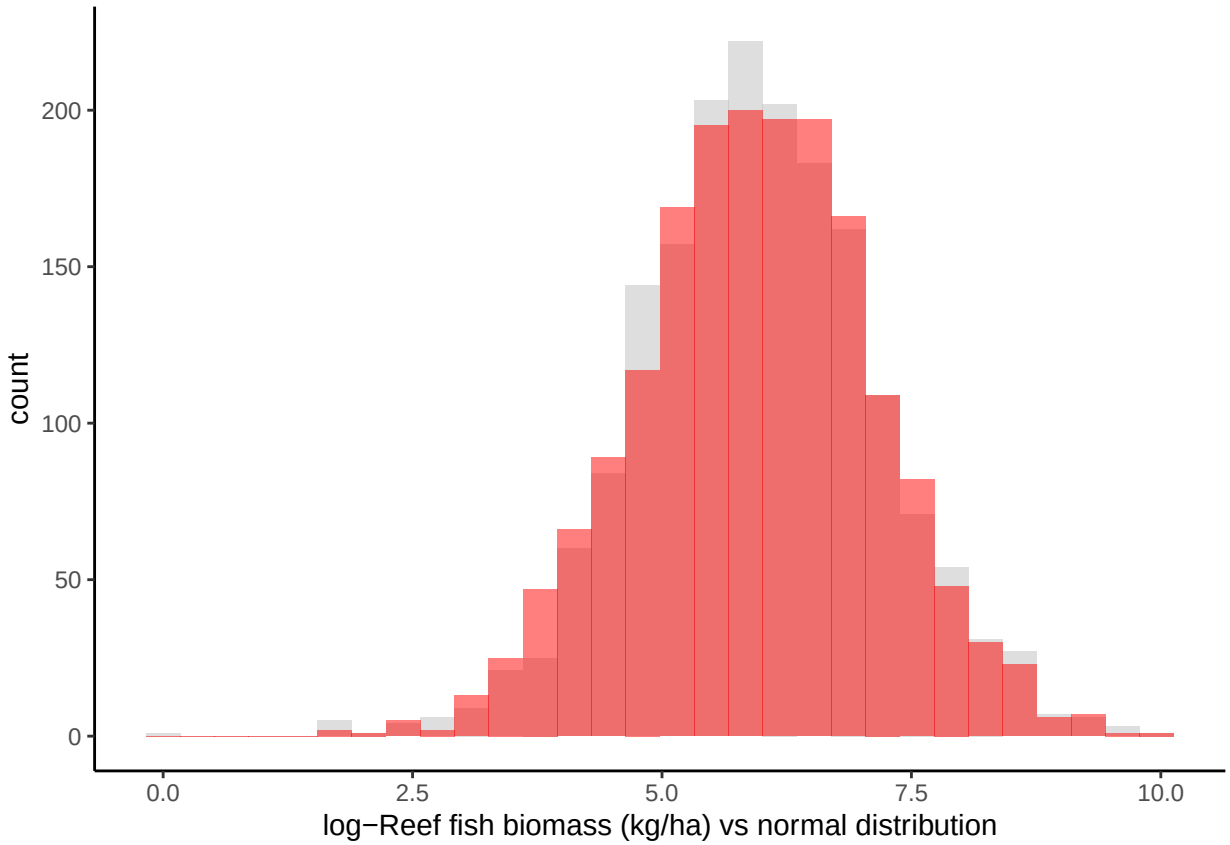


There are several tools that allow us to test if our data is normally distributed.

A first step is to visualize the data (as we have already done), and even contrasting it to simulated data that has the same mean and variance. Go ahead and plot the distribution of log biomass, and overlaid, plot a normal distribution from simulated data that has the same sample size (1798), the mean and the standard of our log-biomass value.

```
#plot log-biomass distribution and a simulated normal distribution with the same mean and standard devi
ggplot()+
  geom_histogram(aes(x=reef_dat$log_biomass),
                 fill="grey",alpha=0.5)+
  geom_histogram(aes(x=rnorm(length(reef_dat$log_biomass),
                             mean(reef_dat$log_biomass),
                             sd(reef_dat$log_biomass))),
                 fill="red",alpha=0.5)+
  theme_classic()+
  xlab ("log-Reef fish biomass (kg/ha) vs normal distribution")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

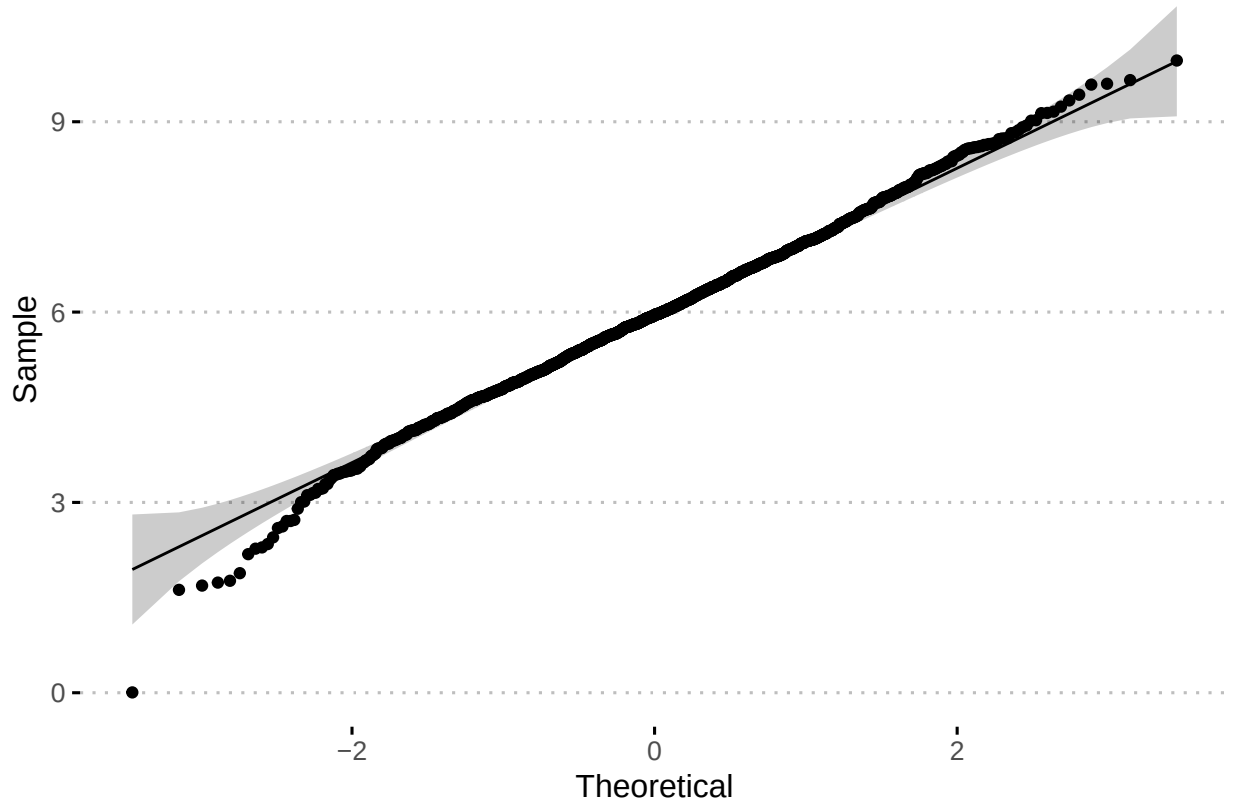


You can see distributions are quite good in terms of overlapping.

Another good visualization to see if your data is normal is a qqplot (quantile to quantile plot). It basically plots the quantiles of our variable to the quantiles of a normal distribution.

```
#qqplot
ggqqplot(reef_dat, x = "log_biomass",
   ggtheme = theme_pubclean())
```

```
## Warning: The following aesthetics were dropped during statistical transformation: sample
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
## The following aesthetics were dropped during statistical transformation: sample
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a ‘group‘ aesthetic or to convert a numerical
##   variable into a factor?
```



In general if points are close to the line and within the confidence interval region, the data is likely normally distributed. This doesn't look to bad for us. The only section that might stand-out is the low biomass values, with a normal distribution slightly underestimating them.

Besides visualizing the data, we can also do a normality test, statistically looking as to whether the data follow the desired distribution. One of these is the Kolmogorov-Smirnov test, with the null hypothesis indicating that the data follow the desired distribution. The Kolmogorov-Smirnov test is useful in that you can compare it to any other distribution you simulated. In R:

```
#ks test
ks.test(reef_dat$log_biomass,
        rnorm(length(reef_dat$log_biomass),
              mean(reef_dat$log_biomass),
              sd(reef_dat$log_biomass)))
```

```
## Warning in ks.test.default(reef_dat$log_biomass,
## rnorm(length(reef_dat$log_biomass), : p-value will be approximate in the
## presence of ties
```

```
##
##  Asymptotic two-sample Kolmogorov-Smirnov test
##
## data:  reef_dat$log_biomass and rnorm(length(reef_dat$log_biomass), mean(reef_dat$log_biomass), sd(re
## D = 0.031702, p-value = 0.3268
## alternative hypothesis: two-sided
```

The Kolmogorov-Smirnov test suggests the data do come from the underlying normal distribution (p.value> 0.05). In general, it is good practice to combine all these visualizations and tests to understand your data distribution, rather than relying on one. Once you fit your models you will also have to check if the underlying distribution you assumed fits well the data or not.

Exploring relationships among variables: Correlations

Often we are interested in how our variable of interest is related to other variables. Correlations are a common way of exploring relationships between multiple variables of interest (e.g., different measurements taken from the same reef site).

A correlation coefficient quantifies the strength and direction of a relationship between two given variables. The correlation coefficient ranges between -1 and 1. 0 corresponds to no relationship, positive coefficient values represent a positive relationship (if one variable increases, so does the other), and negative values represent a negative relationship (if one variable increases, the other decreases).

There are different ways of computing correlations. The default method, the Pearson correlation, assumes that your variables are normally distributed and linearly related. However, there are other methods such as "spearman" or "kendall" that rank the data to compute correlations and tend to be more robust if the data are not normally distributed.

You can compute correlations in R with the "cor()" function. Go ahead and calculate the correlation between log-biomass and human impact (gravtot5002) using different methods.

```
# Pearson correlation
cor(reef_dat$log_biomass, reef_dat$gravtot5002, method = "pearson")
```

```
## [1] -0.1707118
```

```
# Spearman correlation
cor(reef_dat$log_biomass, reef_dat$gravtot5002, method = "spearman")
```

```
## [1] -0.3911843
```

- **gravtot5002**: Is a measure of human impact for each of our reef sites. It is measured as the population size within a 500-km radius of each reef site divided by the travel time (squared) it takes to reach those sites. In other words, reefs with high populations sizes and accessible from those populations will have higher values. In contrast, reefs that do not have any populations nearby will have low values.
- **log-biomass**: Is the log of the measured reef fish biomass in each of those reef sites in kg/ha. It is restricted to reef families that are resident on the reef diurnally and are well captured by Underwater Visual Count sampling methods.

These correlations told us that log-biomass and our human impact metric are negatively correlated.

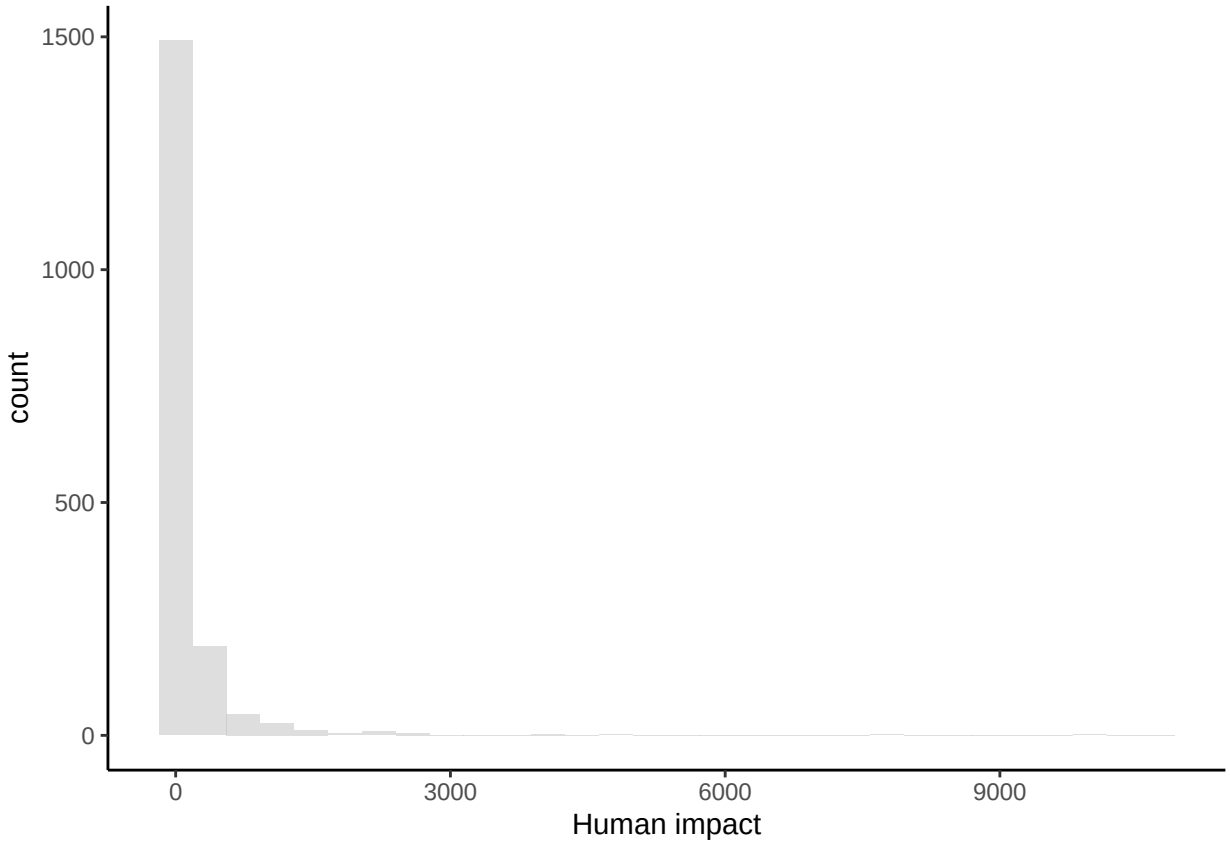
Now do the same, on your own, but making sure the human impact distribution is also normally distributed. To do this, you will likely need to visualize the distribution, check the skewness (is it left or right skewed?) and transform the variable. However, note that this metric does have zeroes so log-transform will not be good (it will make some entries be "-Inf", and delete them from the analyses). Transformations that might help with data that is skewed in this way but has zeroes are double square root (e.g., sqrt(sqrt())) or log(+min(>0)). Go ahead and look at the distribution of log+min human impact. What log+min does, is get the minimum value of your variable of interest that is above zero, and adds that to each entry of your

variable of interest (making the zeroes become the minimum positive values). Once you have transformed those zeros, it applies the log() function.

After looking at the distribution and skewness of the human impact metric, you will have to calculate the min() of gravtot5002, WHERE gravtot5002 is positive. We do this, by restricting our variable to values >0. Then you will have to add that to your gravtot5002 variable and log the result. After that, you will have to estimate the correlations between log biomass an dthe transformed human metric.

```
#plot human impact distribution
ggplot(reef_dat,aes(x=gravtot5002))+geom_histogram(fill="grey",alpha=0.5)+theme_classic()+xlab("Human in
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#calculate skewness
skewness(reef_dat$gravtot5002)
```

```
## [1] 10.3813
```

```
#lowest positive value of human impact
min(reef_dat$gravtot5002[reef_dat$gravtot5002>0])
```

```
## [1] 4.111e-07
```

```
#log of lowest positive value
log(min(reef_dat$gravtot5002[reef_dat$gravtot5002>0]))
```

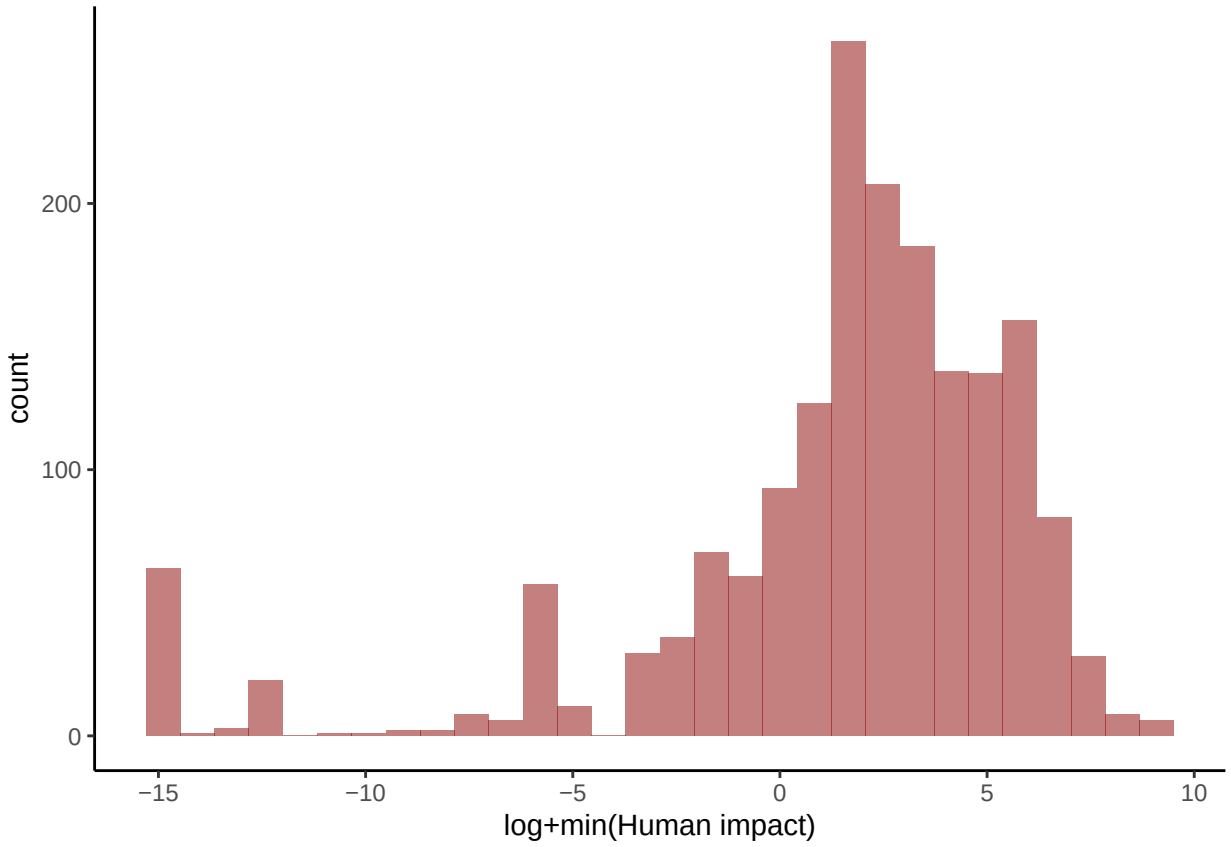
```
## [1] -14.70443
```

```
#human impact distribution on log+min scale
summary(log(reef_dat$gravtot5002+min(reef_dat$gravtot5002[reef_dat$gravtot5002>0])))
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -14.7044   0.1903   2.2700   1.4279   4.3579   9.2801
```

```
#create variable
reef_dat$himpact_logmin<-log(reef_dat$gravtot5002+min(reef_dat$gravtot5002[reef_dat$gravtot5002>0]))
#plot transformed distribution
ggplot(reef_dat,aes(x=himpact_logmin))+geom_histogram(fill="darkred",alpha=0.5)+theme_classic()+xlab("l
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Pearson correlation
cor(reef_dat$log_biomass, reef_dat$himpact_logmin, method = "pearson")
```

```
## [1] -0.3307363
```

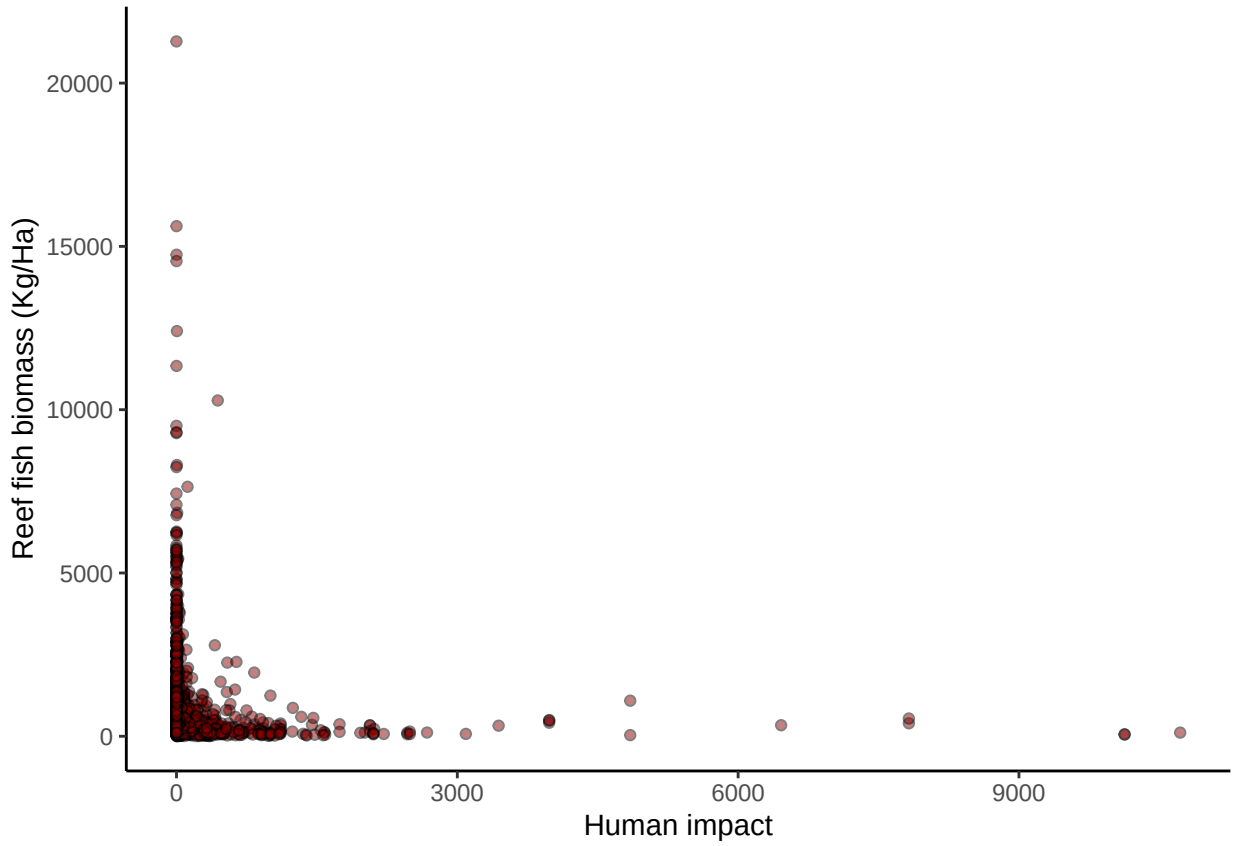
```
# Spearman correlation
cor(reef_dat$log_biomass, reef_dat$himpact_logmin, method = "spearman")
```

```
## [1] -0.3911843
```

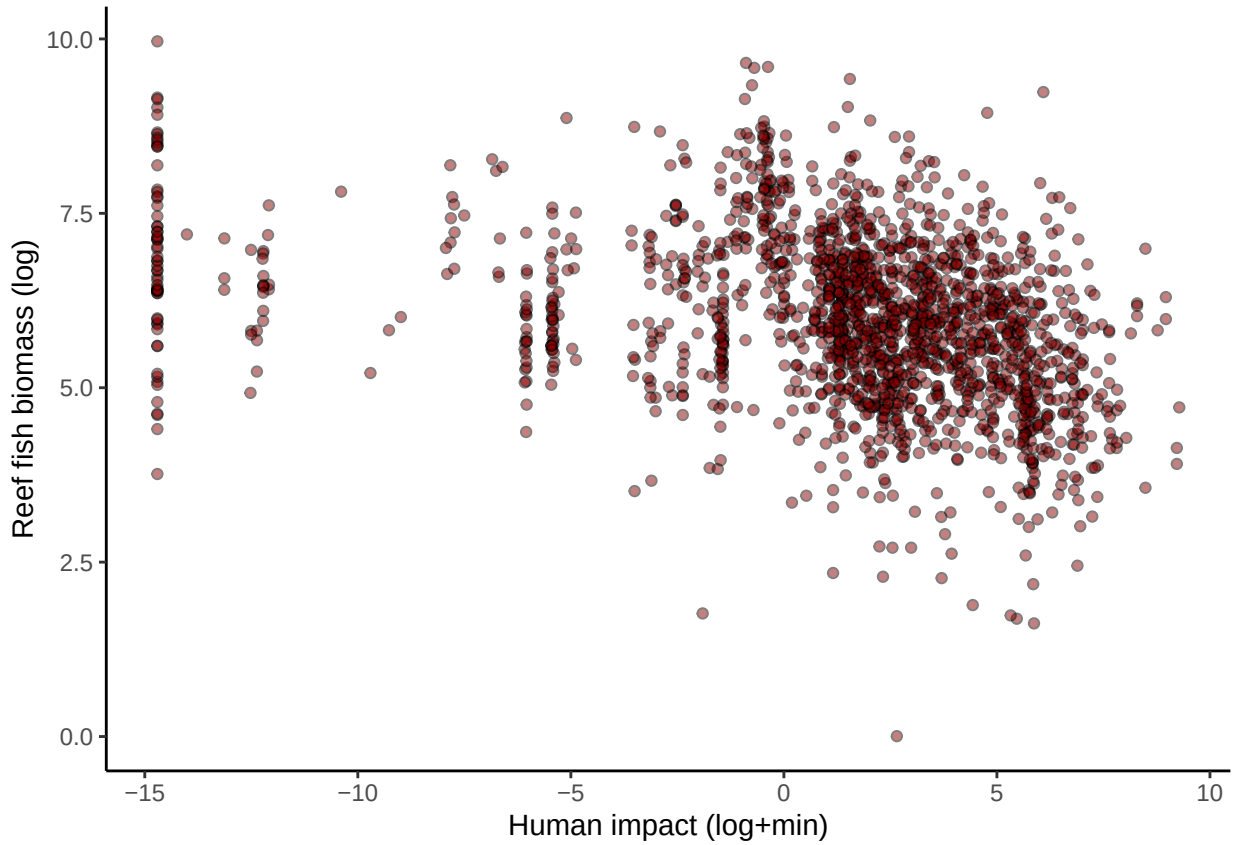
Because the data are more normally distributed, the different correlation tests provide closer results. Overall, consistently, these correlations tell us that biomass and our human impact metric are negatively correlated.

It is always a good idea to visualize the correlations as well as to quantify them because the basic correlation only provides a good description of the relationship if the relationship is ~ linear. So now let's look at the relationship between reef fish biomass (log) and human impact (logmin) with a scatter plot (e.g., geom_point()). Try to do it yourself! Do it with and without transformed variables.

```
#human impact vs biomass
ggplot(reef_dat,aes(x=gravtot5002,y=Biomass...some.families))+geom_point(fill="darkred",pch=21,alpha=0.5
```




```
#human impact vs biomass transformed
ggplot(reef_dat,aes(x=himpact_logmin,y=log_biomass))+geom_point(fill="darkred",pch=21,alpha=0.5)+theme_
```



QUESTION: Do you think human impact is correlated with reef fish biomass? if so how? How has transforming the data helped?

Yes, we see that reefs only have high biomass values at low human impact. In other words, all high human impact locations have relatively low biomass values. Transforming the data, besides allowing us to make variables closer to normal distributions, has allowed us to see the patterns better. We see that there is lots of noise and this is because there are a lot of factors that we have not accounted for that are different among reef sites (e.g., depth, habitat types...). However, on average, biomass stays fairly constant at low human impact values (<0 in transformed units), and as human impact increases, the biomass tends to decrease.

Generalized linear models: simple regression

OK, we know how to estimate correlations, but what if we wanted to predict the biomass of a site given its human impact? and do that accounting for other confounding effects like the type of reef habitat? One way to do that is with statistical models.

Today I am going to explain Generalized linear (mixed) models to quantify the association between a response (outcome/dependent) variable and the predictor/s (covariate/s,independent variable/s). These models can be used for almost any data and study design, but they take assumptions to simplify the complex reality of nature. So, remember: VERY MODEL IS WRONG, BUT SOME ARE USEFUL.

We have gone through in the class the structure of a generalized linear model (power point presentation). Now, let's put it into practice. We are going to start simple, but note this dataset has a lot of different layers, so we are going to subset it for the purpose of understanding the models.

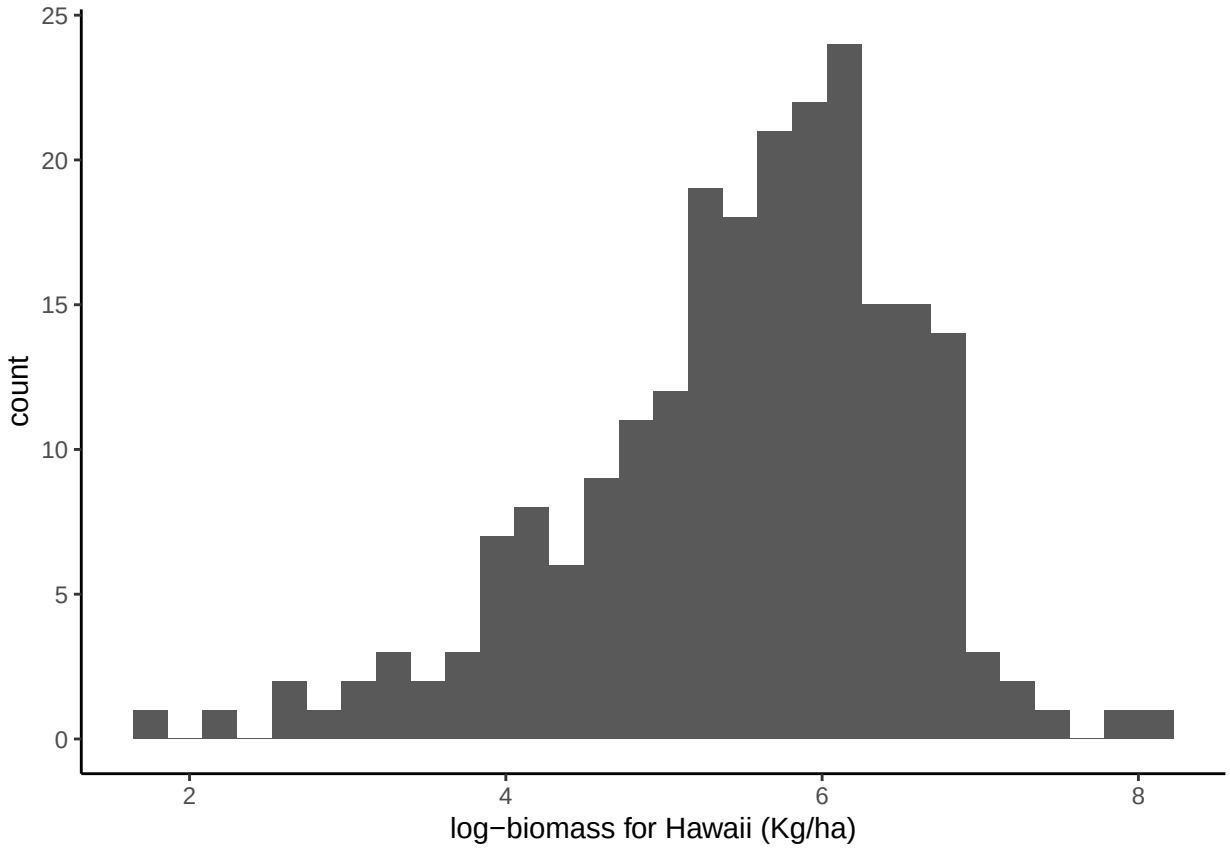
Go ahead and filter the reef_dat so it only includes the sites in Hawaii (Note that the variable "Larger" is the one that house the country names).

```
#obtain only Hawaii data
hi_dat<-reef_dat[reef_dat$Larger=="Hawaii",]
#eliminate other levels
hi_dat<-droplevels(hi_dat)
```

For this section, we are going to focus on biomass (log-biomass for simplicity given that we have already checked that it follows a normal distribution). Remember it is a continuous variable. Inspect the log-biomass variable for this data subset.

```
ggplot(hi_dat,aes(x=log_biomass))+geom_histogram()+ theme_classic()+xlab("log-biomass for Hawaii (Kg/ha
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

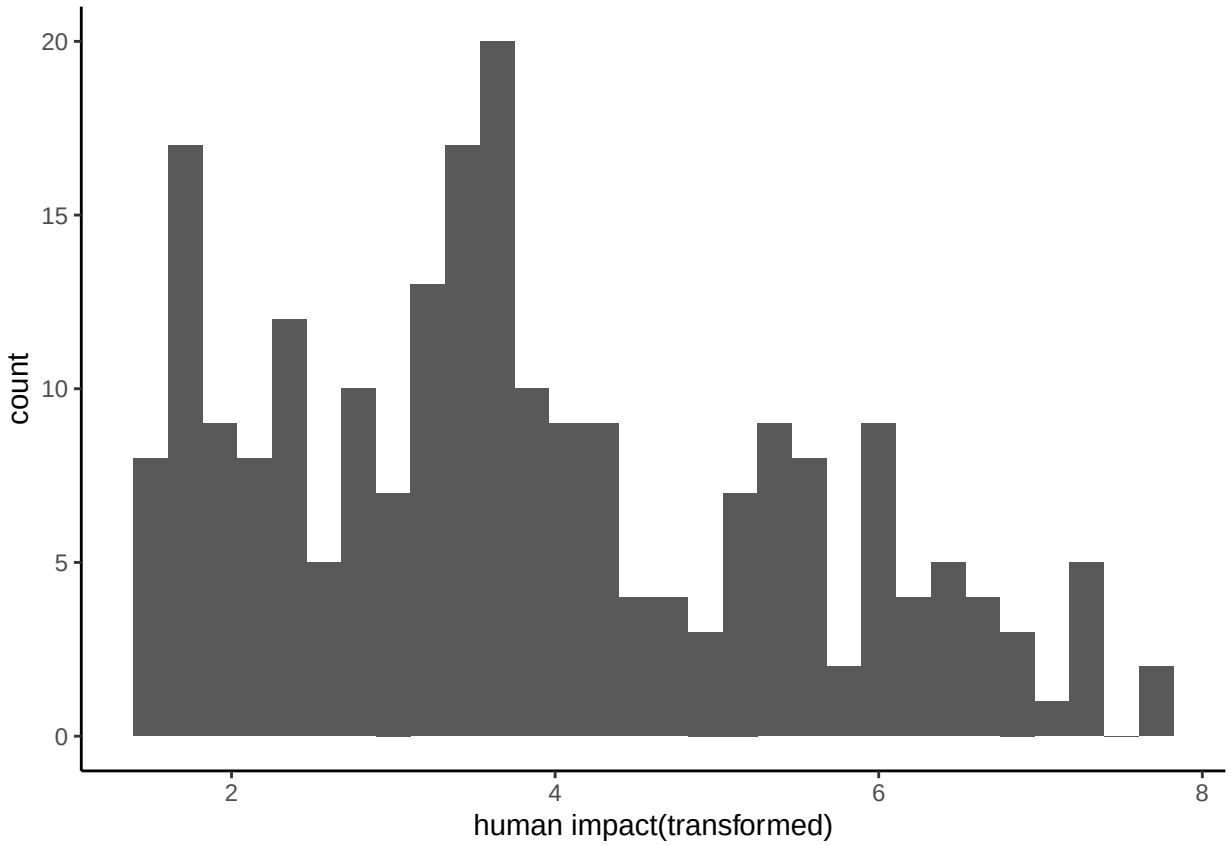


We can see that, in Hawaii, reef sites span high and low biomass values, and that the log-biomass does tend to follow a normal distribution.

Now do the same for the transformed human metric. What can you tell me about the human impact in our sampled Hawaii reefs?

```
ggplot(hi_dat,aes(x=himpact_logmin))+geom_histogram()+ theme_classic()+xlab("human impact(transformed)"
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

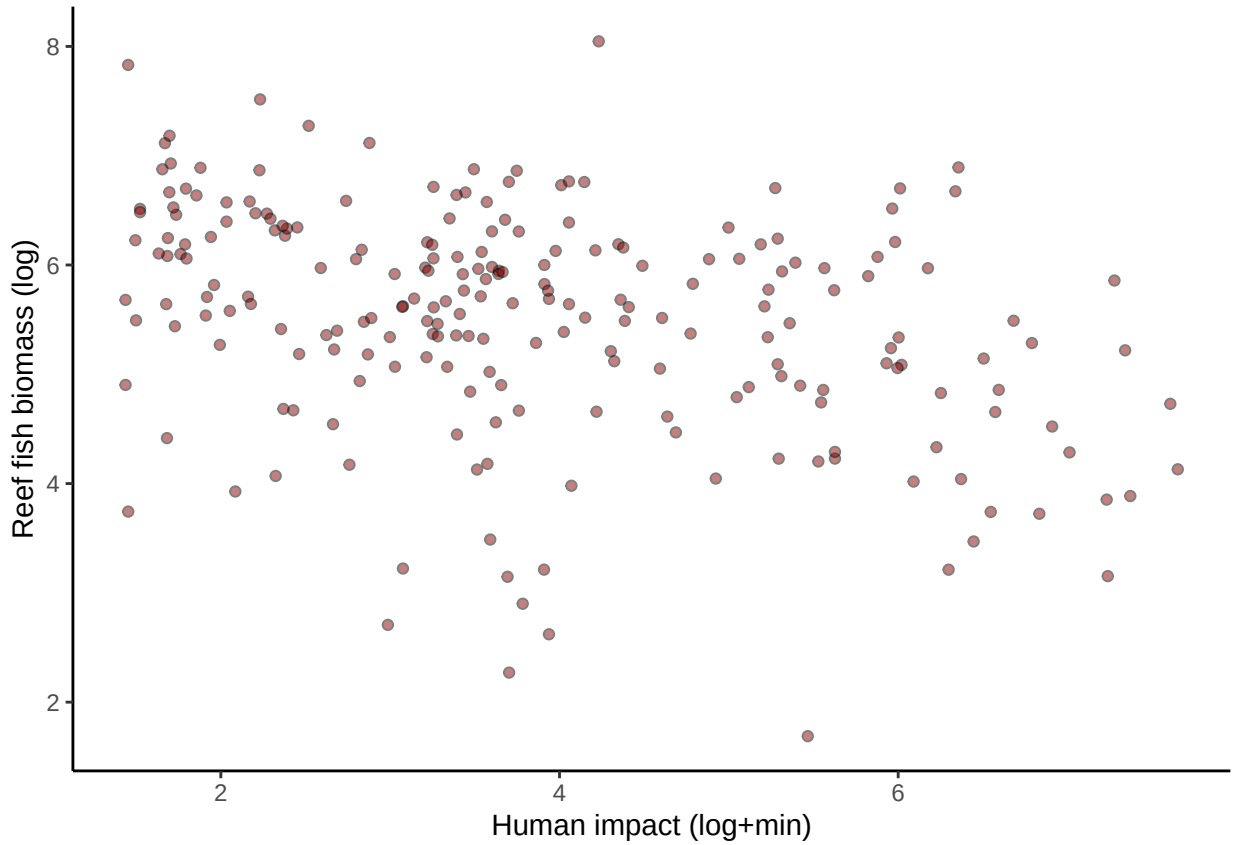


We see that sampling in Hawaii covered a gradient of human impact values, some sites that have very low human impact, and others with higher values.

If this was the only data we had collected and we wanted to predict biomass from human impact we could run a simple linear model. We will do this in R using the glm() or lm () function already built within R:

However, the first thing is to visualize how the variables are related. You have already done this for the entire dataset. Go ahead and do it for this Hawaii subset:

```
#human impact vs biomass transformed
ggplot(hi_dat,aes(x=himpact_logmin,y=log_biomass))+geom_point(fill="darkred",pch=21,alpha=0.5)+theme_cla
```



Here each point is a site. After inspecting the data, We would say that sites with high human impact tend to have lower biomass values than those with low human impact.

We want to fit a linear model to this relationship. We can do this with lm() function from base R or glm() function (specifying the family distribution of our response variable-which is gaussian in this example)

```
#simple linear model of biomass
lm_biomass<-lm(log_biomass~himpact_logmin,data=hi_dat)
print(lm_biomass)
```

```
##
## Call:
## lm(formula = log_biomass ~ himpact_logmin, data = hi_dat)
##
## Coefficients:
##    (Intercept)  himpact_logmin
##         6.4000         -0.2324
```

```
#alternatively, We could also do it using "glm" which also allows for other error distributions besides
glm_biomass<-glm(log_biomass~himpact_logmin,data=hi_dat, family="gaussian")
print(glm_biomass)
```

```
##
## Call:  glm(formula = log_biomass ~ himpact_logmin, family = "gaussian",
##     data = hi_dat)
##
```

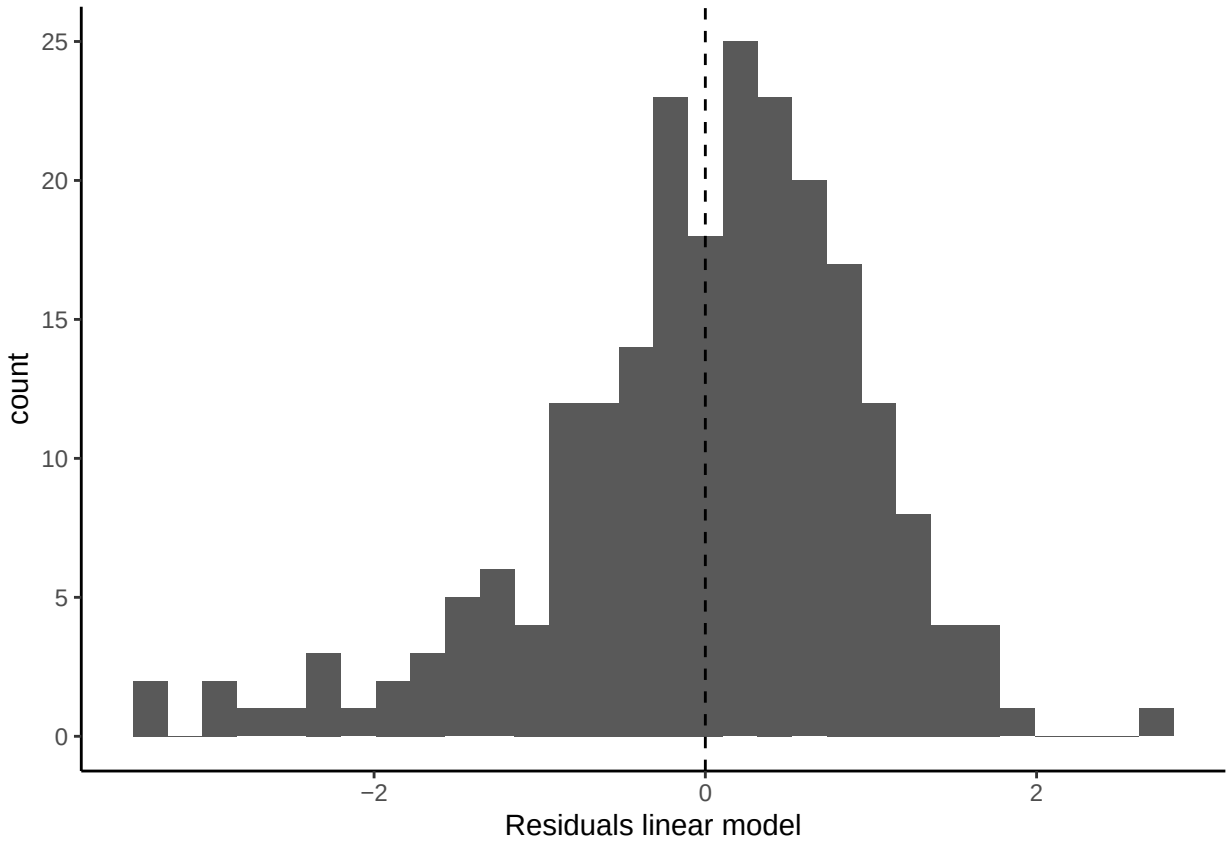
```
## Coefficients:
##    (Intercept)  himpact_logmin
##         6.4000         -0.2324
##
## Degrees of Freedom: 223 Total (i.e. Null);  222 Residual
## Null Deviance:       241.4
## Residual Deviance: 211.1     AIC: 628.4
```

This is telling us that the average log-biomass in the absence of human impact for Hawaii is ~6.4, and the effect of the transformed human impact on log-biomass is ~-0.23. So for every increase in the transformed human impact, based on association, we expect around 0.23 decrease in log-biomass.

So we fitted a model to our data. However, the first thing to do, is to test whether the model fits well our data. Let's do that now. The first thing we are going to do is check the residual error. This is telling us how our observations differ from our expectations or predictions. Because we assumed a normal distribution, we expect the residuals to be normally distributed around 0. You already knw hot to plot a distribution. However, now your x axes should be the residuals e.g., resid(lm_biomass)

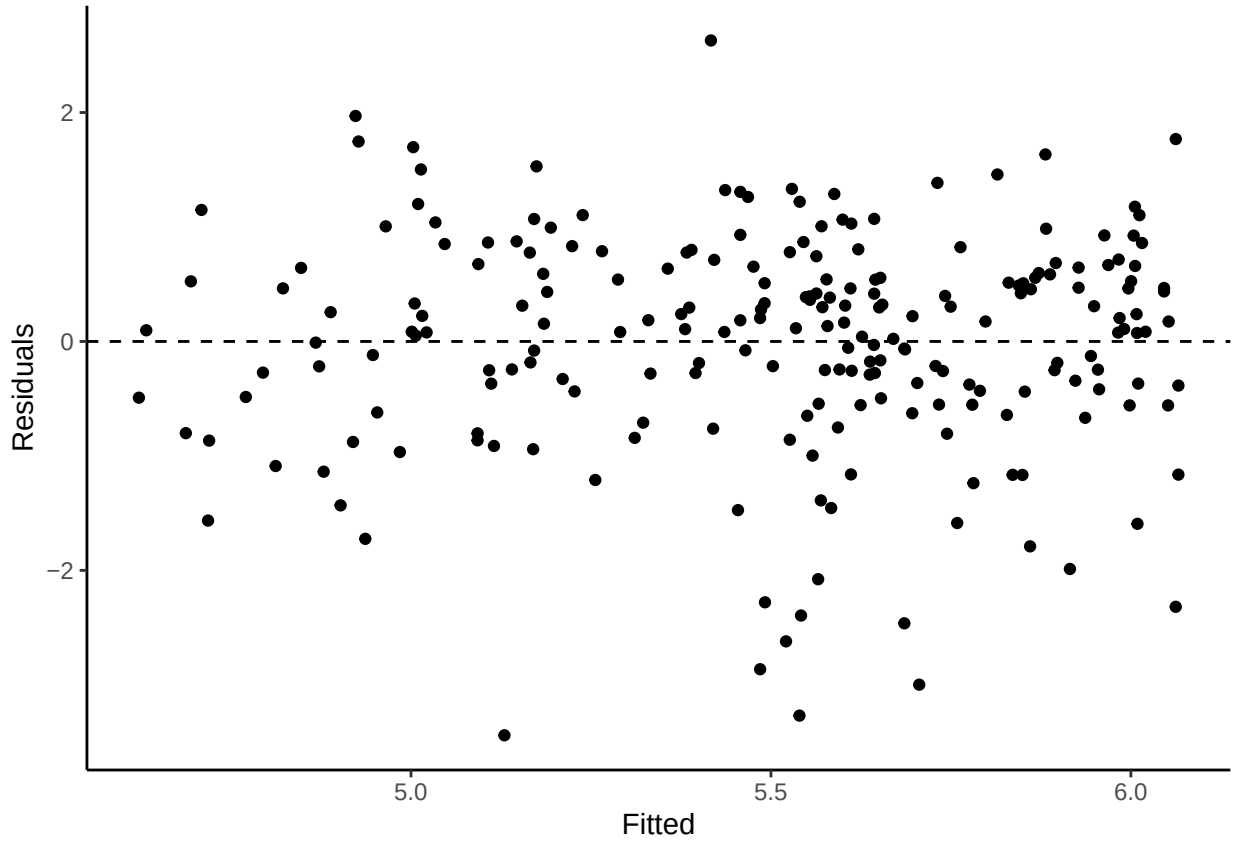
```
#plot distribution of residuals
ggplot()+geom_histogram(aes(x=resid(glm_biomass)))+ theme_classic()+xlab("Residuals linear model")+geom_
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Another useful plot to test that our model fits well our data is to do a fitted vs residuals plot. Fitted is the model predictions, and the residuals are the observed minus the fitted values.

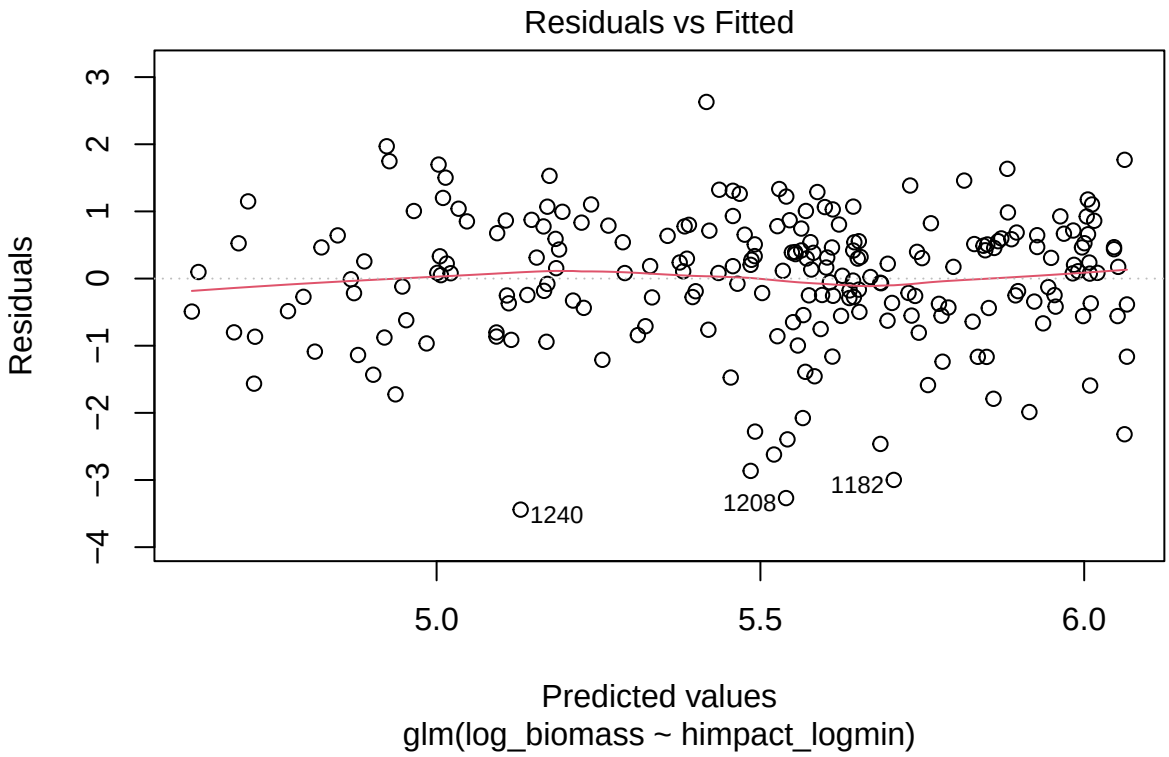
```
#plot residuals vs fitter
ggplot()+geom_point(aes(y=resid(glm_biomass), x=fitted(glm_biomass)))+ theme_classic()+ylab("Residuals")
```

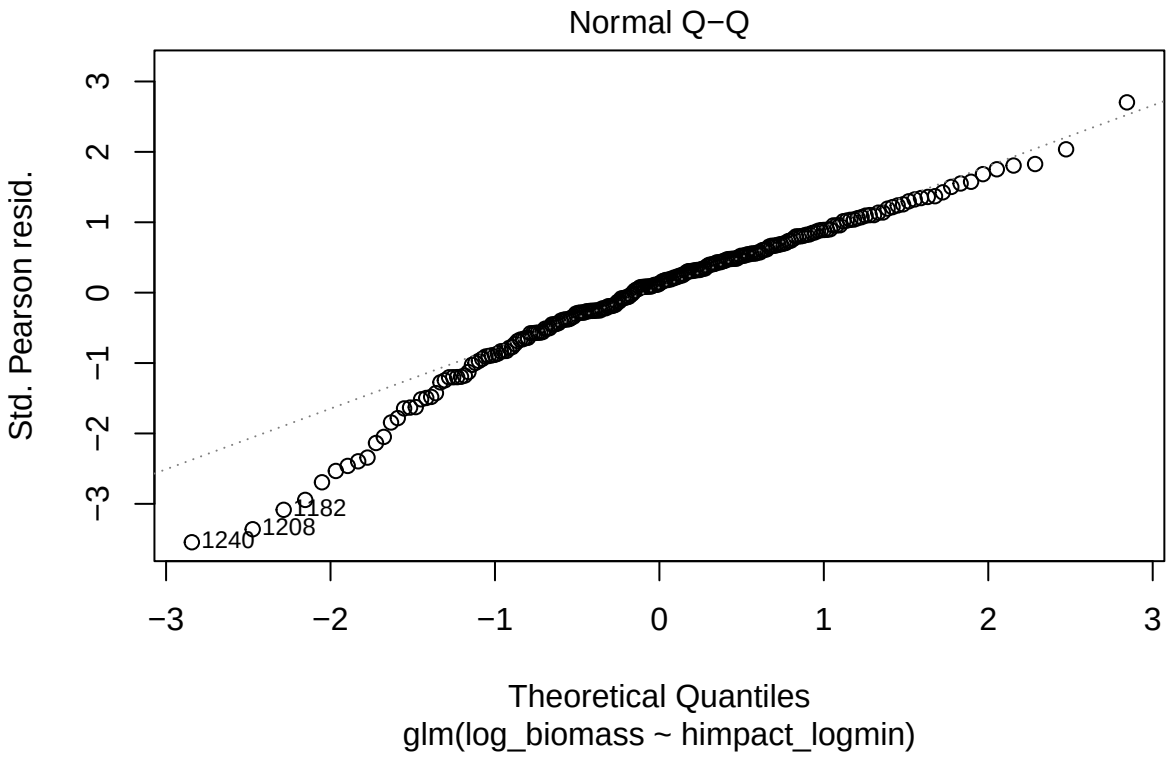


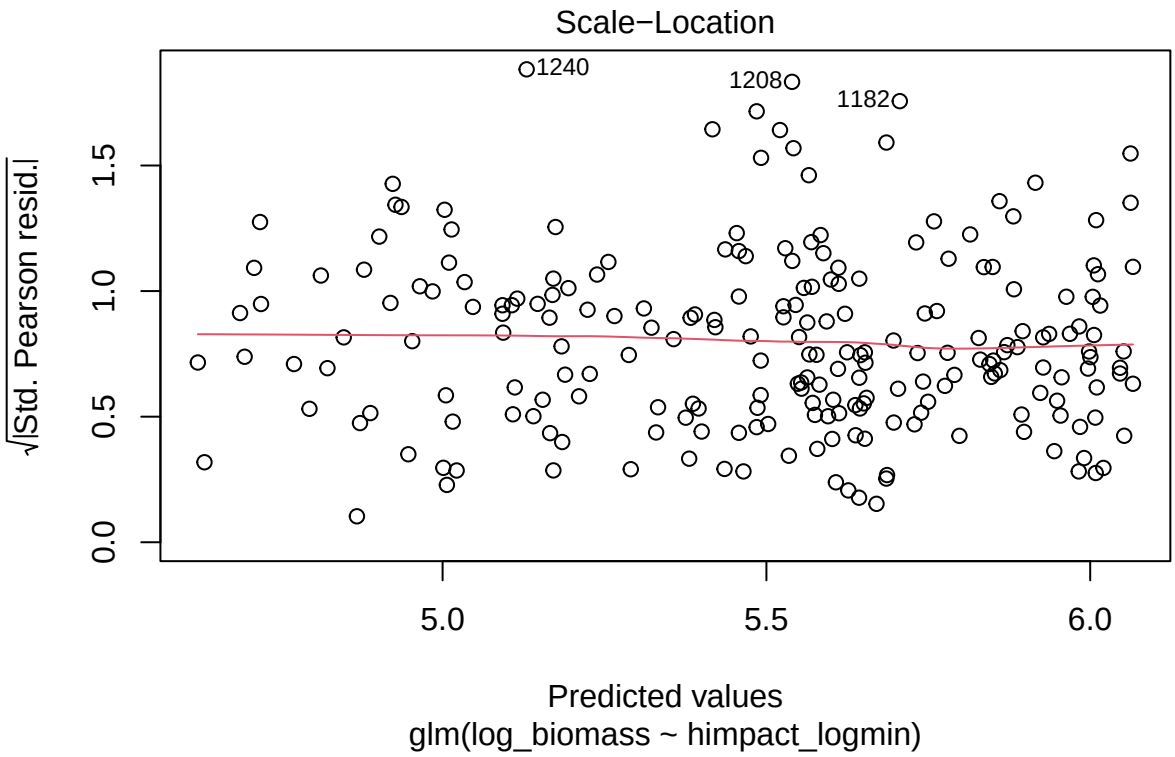
Here each point is a site. The x axes is what the model predicted the site's biomass is, whereas the y axes is how that fitted values diverged from the observed values.

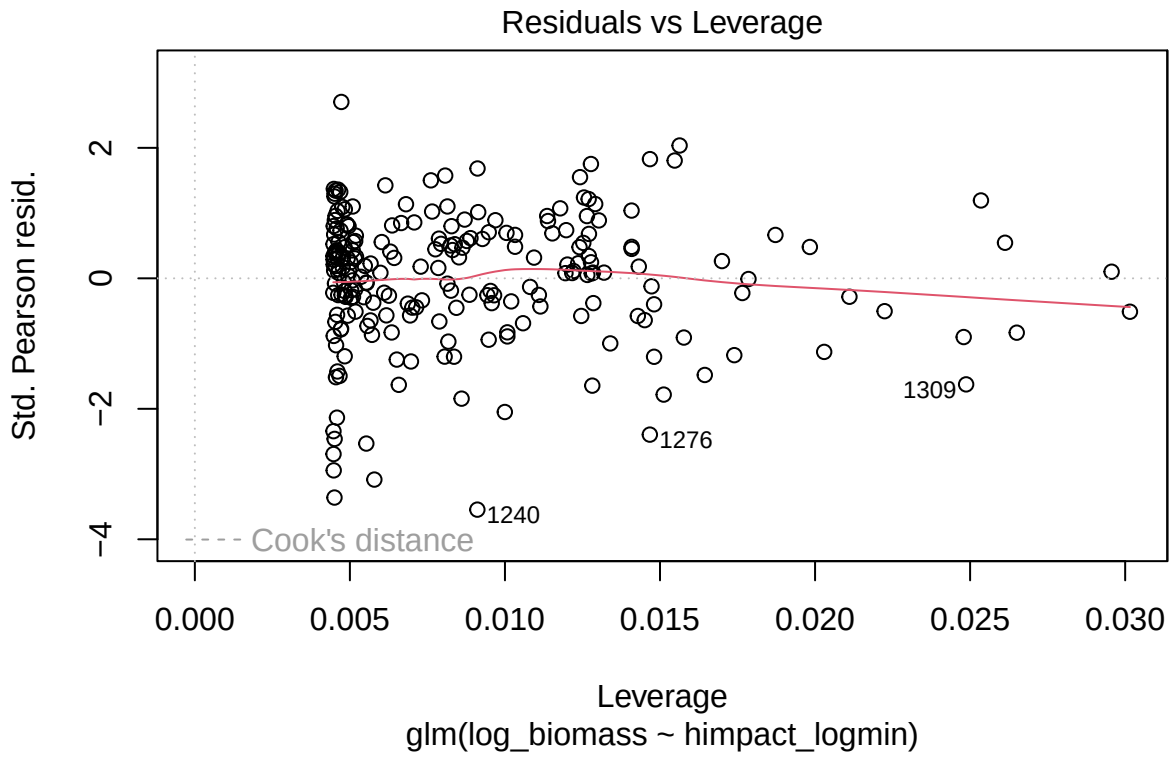
Another option is to use the plot() function with the model, which gives you model fit results. Try it out!

```
#plot fit diagnostics
plot(glm_biomass)
```


Residuals vs Fitted
Residuals
3
2
1
0
−1
−2
−3
−4
1240
1208
1182
5.0
5.5
6.0
Predicted values
glm(log_biomass ~ himpact_logmin)


Normal Q−Q
Std. Pearson resid.
Theoretical Quantiles
glm(log_biomass ~ himpact_logmin)
1240
1208
1182


Scale−Location
1240
1208
1182
√|Std. Pearson resid.|
0.0
0.5
1.0
1.5
5.0
5.5
6.0
Predicted values
glm(log_biomass ~ himpact_logmin)



Some may seem familiar (e.g., quantile to quantile plot with assumed distribution). The only one that may not is the leverage, which is saying how influential each observation is on the model fit. None of our observations are big outlines or driving the model results. which is good.

Now we have a good fitted model,we can start to using the results:

QUESTION: using R, tell me how much biomass (in kg/ha-raw values) a reef in Hawaii is expected to have in the absence of human impact? Remember that the opposite to log() is exp().

```
#mean expected biomass in Hawaii in the absence of human impact (i.e., intercept) in kg/ha
exp(glm_biomass$coefficients[1])
```

```
## (Intercept)
##    601.8159
```

We expect the biomass to be 601.8 kg/ha.

QUESTION: now tell me how much raw biomass(in kg/ha) we would expect if the transformed human impact were 3.2? Remember the equation and remember that we transformed the variables.

```
#mean expected biomass in Hawaii (in kg/ha) when transformed human impact is 3.2
exp(glm_biomass$coefficients[1]+glm_biomass$coefficients[2]*3.2)
```

```
## (Intercept)
##    286.1096
```

We expect the biomass to be ~286 kg/ha.

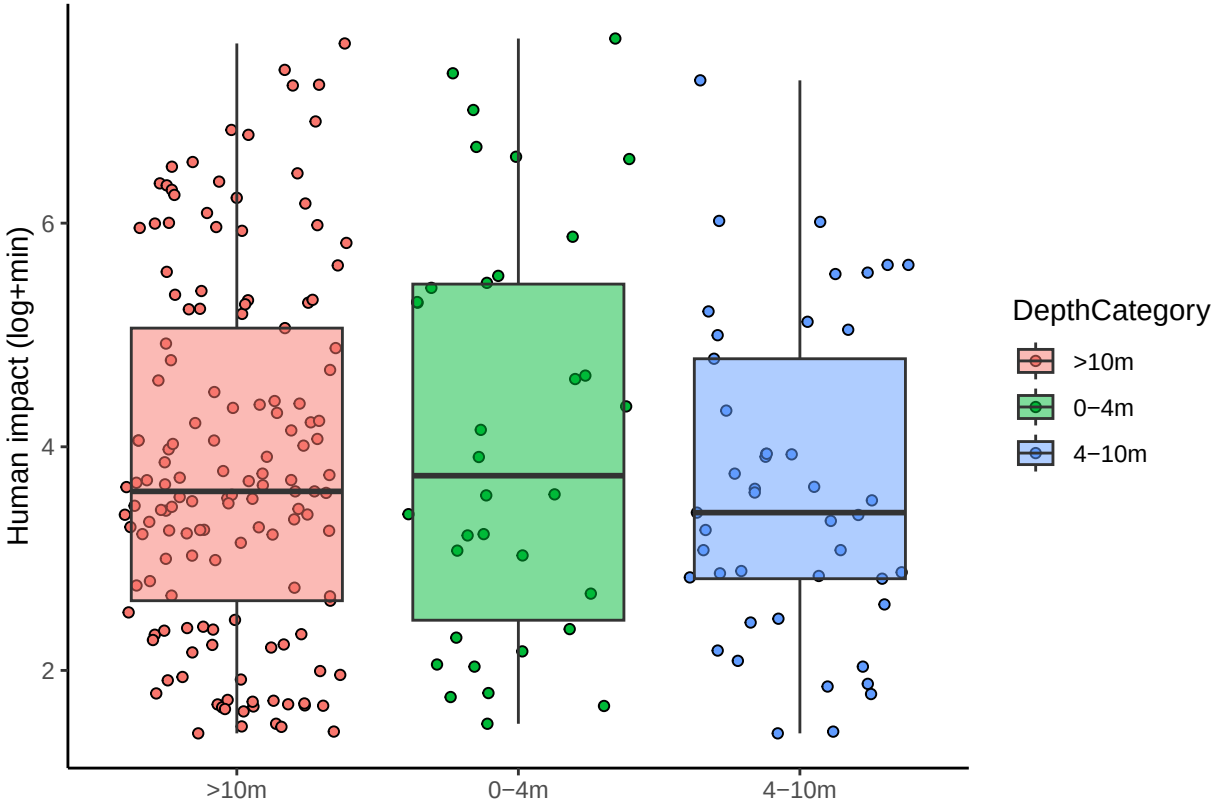
Well done! You can basically now predict biomass from the level of human impact! Note thought, that while these models are very helpful to make inferences, they do not imply causation, but instead association (there could be other variables correlated to our human impact metric - e.g., contamination, market access- that could be driving biomass.

Generalized linear models: multiple regression

Many times, we do not have only one predictor or covariate such as human impact. Sometimes, even if our research question is about how human impact is associated with biomass, we want to control for things that we have measured and we know have an impact on our response variable (e.g., depth category- the depth at which sampling was performed). This is when we multiple linear regression becomes handy. Look at the slides on this!

The first ting to check if how correlated your potential predictors are and how they are related. If predictors were continuous, a good function to check correlations would be pairs() (feel free to look at the help function to see what this function does in your own time). For categorical variables like DepthCategory, it is sometimes useful to use boxplots or violin plots. Go ahead and plot the transformed human impact variable against depth category with a boxplot:

```
#plot potential covariates
ggplot(hi_dat,aes(x=DepthCategory,y=himpact_logmin))+geom_jitter(aes(fill=DepthCategory),pch=21)+geom_b
```



There are no major trends between human impact and depth category so we can go ahead and create a new model that predicts log-biomass from both human impact and depth category:

```r
#multiple linear model of biomass
glm_biomass_hab<-glm(log_biomass~himpact_logmin+DepthCategory,data=hi_dat, family="gaussian")
print(glm_biomass_hab)
```

```
##
## Call:  glm(formula = log_biomass ~ himpact_logmin + DepthCategory, family = "gaussian",
##     data = hi_dat)
##
## Coefficients:
##        (Intercept)      himpact_logmin   DepthCategory0-4m  DepthCategory4-10m
##             6.4901             -0.2299             -0.3753             -0.2118
##
## Degrees of Freedom: 223 Total (i.e. Null);  220 Residual
## Null Deviance:       241.4
## Residual Deviance: 206.5     AIC: 627.4
```

Assuming the model fits well to our data, this is telling us that the predicted log-biomass of a site is relted to depth category and human impact by the following formula: log_biomass=6.49 - 0.23transformed_humanimpact - 0.38depth04m - 0.21depth410m. In other words, shallow depths and human impact have a negative association with biomass. The reference value for depth is >10m. In other words, if the depth is >10m and the transformed human impact is 0, the expected log-biomass would be ~6.5.

QUESTION: what is the predicted RAW biomass in kg/ha if human impact (transformed) is 0 and the sites was sampled from 0 to 4 m?

```r
#mean expected biomass in Hawaii (in kg/ha) when transformed human impact is 0 and site's depth is from
exp(glm_biomass_hab$coefficients[1]+glm_biomass_hab$coefficients[2]*0+glm_biomass_hab$coefficients[3]*1
```

```
## (Intercept)
##    452.5141
```

```r
#or
exp(glm_biomass_hab$coefficients[1]+glm_biomass_hab$coefficients[3])
```

```
## (Intercept)
##    452.5141
```

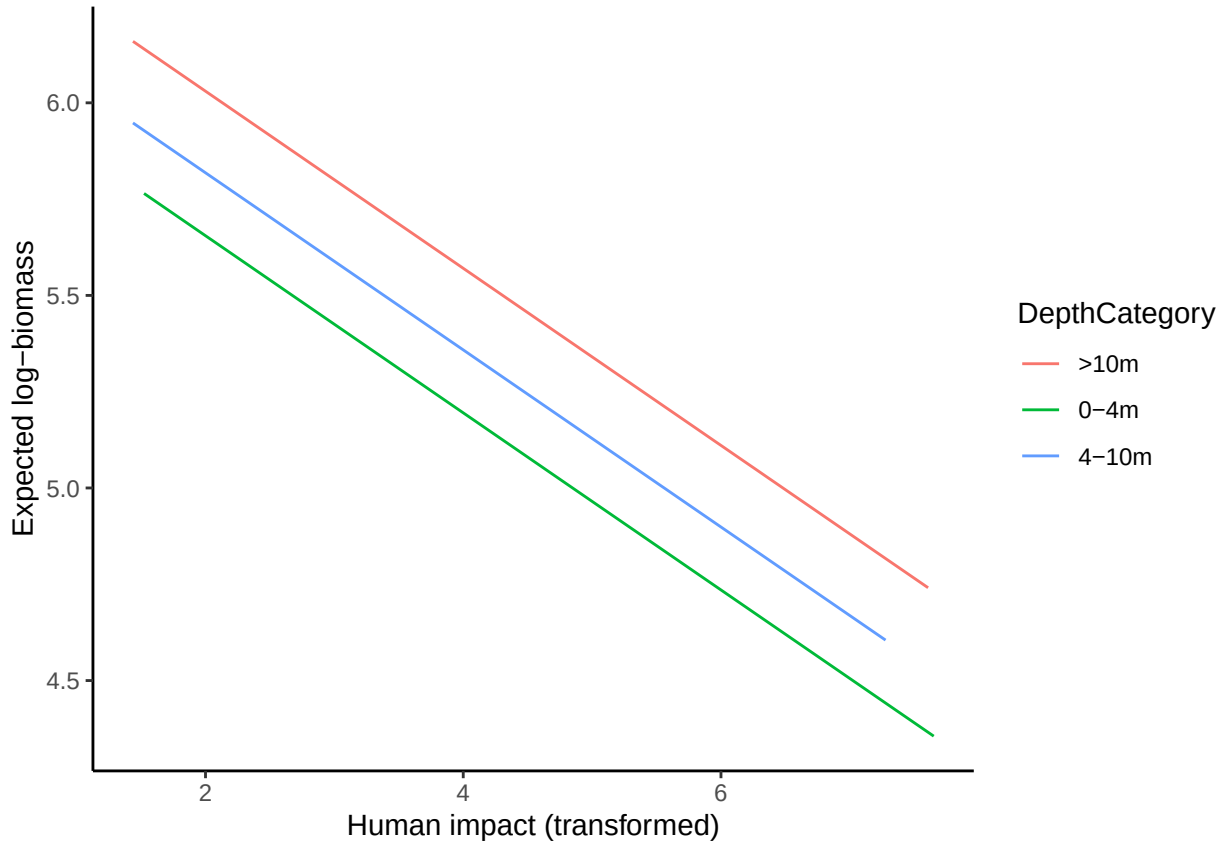
The expected biomass is ~452.5 kg/ha.Cogratulations! You just predicted biomass. You can further predict log-biomass using the predict() function. For example, if we wanted to predict biomass for our sues, given their human impact and depth sampling conditions:

```r
#predict log-biomass
predict(glm_biomass_hab)
```

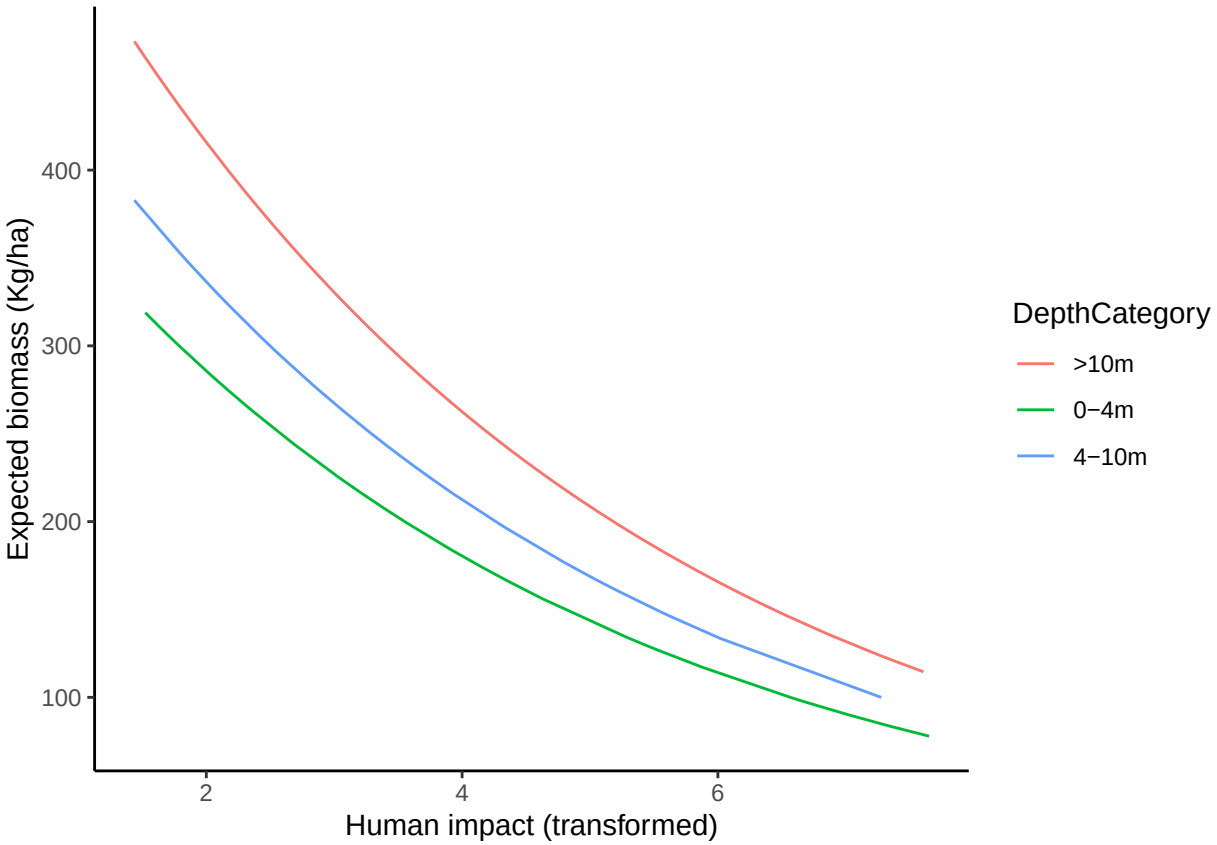
```
##     1016     1017     1018     1019     1020     1021     1022     1023
## 5.367401 5.644301 5.457795 5.372836 5.633894 5.440835 5.575582 5.378981
##     1024     1025     1026     1027     1028     1029     1030     1031
## 5.286589 5.976895 5.977772 6.049014 5.943090 5.940355 5.956883 5.846615
##     1032     1033     1034     1035     1036     1037     1038     1039
## 5.500677 5.602216 5.741114 5.571075 5.564510 5.993202 6.092609 6.043721
##     1040     1041     1042     1043     1044     1045     1046     1047
```

```
## 5.628387 5.297074 5.627105 5.287575 5.676065 5.749963 5.493927 5.662059
##     1048     1049     1050     1051     1052     1053     1054     1055
## 5.876350 5.669105 5.719918 5.216136 5.735908 5.624151 5.568007 5.497105
##     1056     1057     1058     1059     1060     1061     1062     1063
## 5.682650 5.374712 5.686635 5.413765 5.724749 5.160308 4.598676 5.590950
##     1064     1065     1066     1067     1068     1069     1070     1071
## 5.177495 5.521601 5.860124 5.662059 5.967703 5.946291 5.615843 5.810611
##     1072     1073     1074     1075     1076     1077     1078     1079
## 5.647207 5.983158 5.886848 5.510979 6.031551 5.798714 5.653222 5.851488
##     1080     1081     1082     1083     1084     1085     1086     1129
## 6.102817 5.948540 5.570006 5.587578 5.642788 5.709719 6.039436 4.901376
##     1130     1131     1132     1133     1134     1135     1136     1137
## 5.490378 5.629762 5.326315 4.894083 5.520100 5.698086 5.910895 5.955751
##     1138     1139     1140     1141     1142     1143     1144     1145
## 5.640941 5.008044 5.358144 5.284135 5.926371 5.877911 5.855607 5.777557
##     1146     1147     1148     1149     1150     1151     1152     1153
## 5.712061 5.616576 5.800711 5.571009 5.673748 4.763182 5.408666 5.517105
##     1154     1155     1156     1157     1158     1182     1183     1184
## 5.112062 5.028640 5.117852 5.129035 5.000406 5.803396 5.794170 5.767782
##     1185     1186     1187     1188     1189     1190     1191     1192
## 5.719645 5.701647 5.750854 5.625543 5.709592 5.682358 5.444978 5.374084
##     1193     1194     1195     1196     1197     1198     1199     1200
## 5.693822 5.498749 5.742469 5.618734 5.614033 5.476293 5.377391 6.139912
##     1201     1202     1203     1204     1205     1206     1207     1208
## 6.100126 6.106144 5.764661 6.102132 5.483780 5.412352 5.748382 5.638766
##     1209     1210     1211     1212     1213     1214     1215     1216
## 5.032777 4.984507 5.114611 5.111378 5.269058 5.257756 5.250021 5.274069
##     1217     1218     1219     1220     1221     1222     1223     1224
## 5.554253 5.647650 5.481595 5.392341 4.896032 4.918811 5.433996 5.055815
##     1225     1226     1227     1228     1229     1230     1231     1232
## 5.048711 5.277357 5.536714 5.101480 4.984507 5.126265 4.578513 4.898818
##     1233     1234     1235     1236     1237     1238     1239     1240
## 5.268186 4.897479 5.197402 4.994418 4.355612 4.843648 4.868316 4.858000
##     1241     1242     1253     1254     1255     1256     1257     1258
## 5.210659 5.295072 5.557521 5.557521 5.557521 5.620308 5.649507 5.665215
##     1259     1260     1261     1262     1263     1264     1265     1266
## 5.691664 5.372862 5.700156 5.735450 5.743136 5.741477 5.529637 5.333692
##     1267     1268     1269     1270     1271     1272     1273     1274
## 5.710107 5.418553 5.452570 5.468728 5.677319 5.292972 5.639108 6.099857
##     1275     1276     1277     1278     1279     1280     1281     1282
## 5.944221 6.155950 6.104359 6.159631 6.145394 6.090903 6.098179 6.114516
##     1283     1284     1285     1286     1287     1288     1289     1290
## 5.701561 6.094632 5.846009 6.077560 5.728100 6.146259 5.866952 5.947784
##     1291     1292     1293     1294     1295     1296     1297     1298
## 6.109629 6.050637 4.795345 5.080178 5.118503 5.070112 4.740981 5.058449
##     1299     1300     1301     1302     1303     1304     1305     1306
## 5.120299 5.042020 5.109895 5.025116 5.052481 4.604984 4.827399 4.928896
##     1307     1308     1309     1310     1311     1312     1313     1314
## 5.089558 5.003281 4.825893 4.603381 4.984809 4.502558 4.427071 5.151164
```

```
#plot expected biomass given sites human impact and depth
ggplot()+geom_line(data=hi_dat,aes(y=predict(glm_biomass_hab),x=himpact_logmin,col=DepthCategory))+ylab
```




```
#alternatively if we wanted to plot the expected RAW biomass
ggplot()+geom_line(data=hi_dat,aes(y=exp(predict(glm_biomass_hab)),x=himpact_logmin,col=DepthCategory))+
```

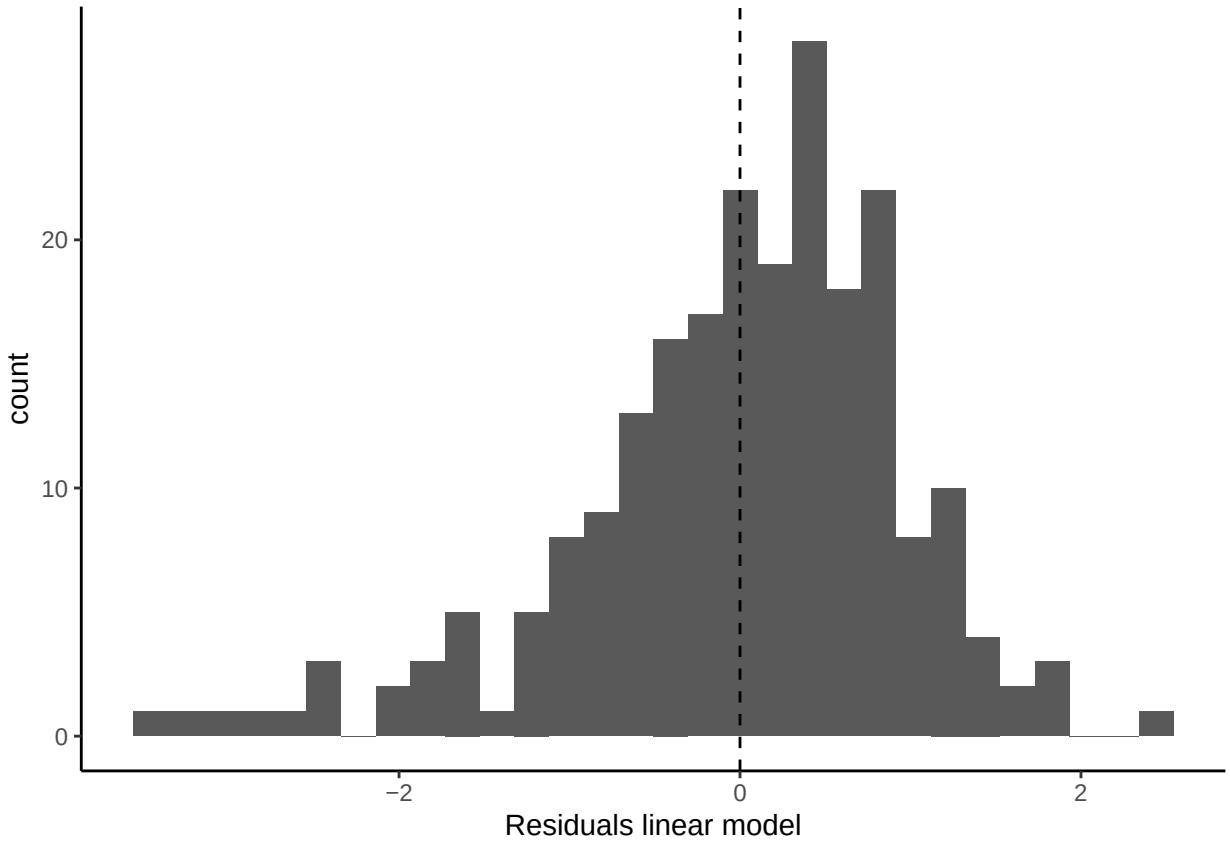


Given our model assumptions, we expect biomass to decrease in with increased human impact. However, we expect larger depths to have greater biomass of reef fish.

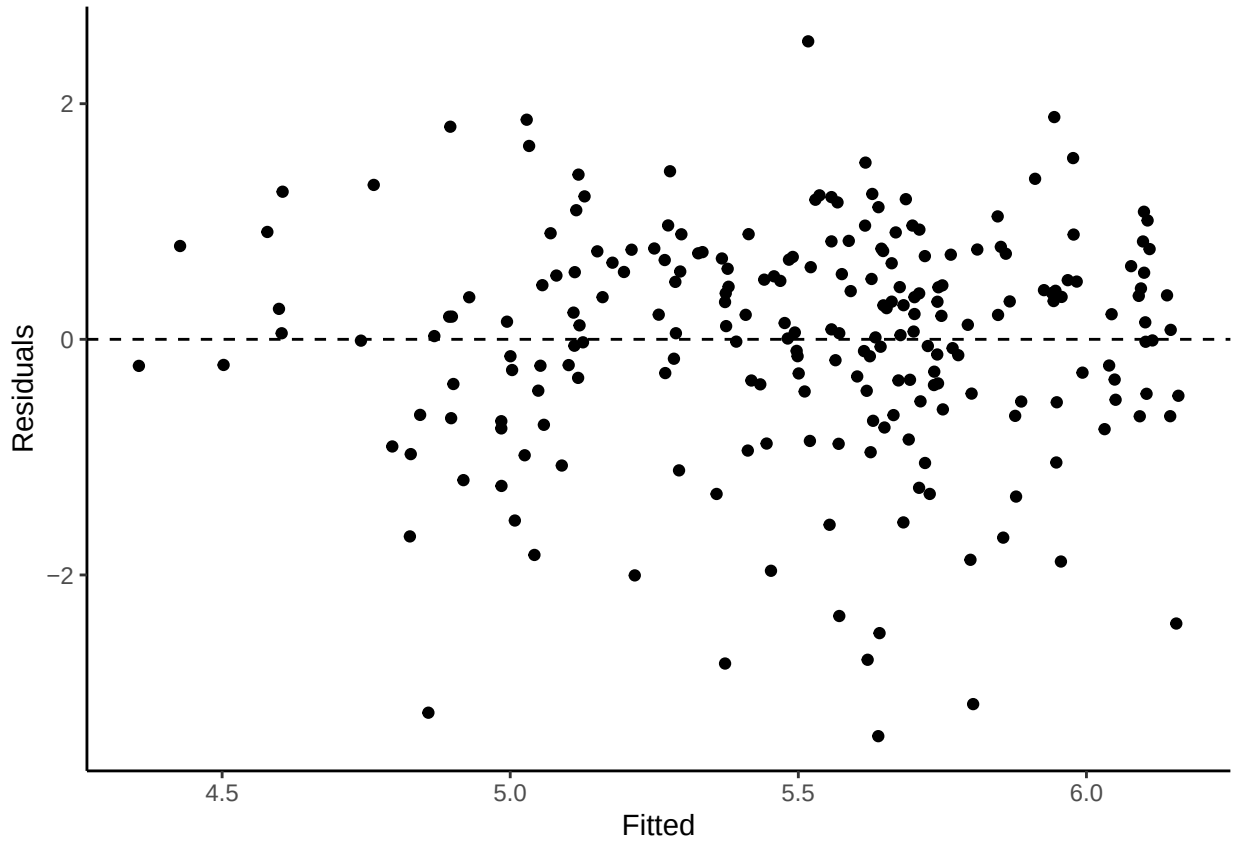
Note, we assumed that the model fit our data well. However, we did not check it! On your own, following the model checking we did above, check that our model fit well our data.

```
#plot distribution of residuals
ggplot()+geom_histogram(aes(x=resid(glm_biomass_hab)))+ theme_classic()+xlab("Residuals linear model")+
```

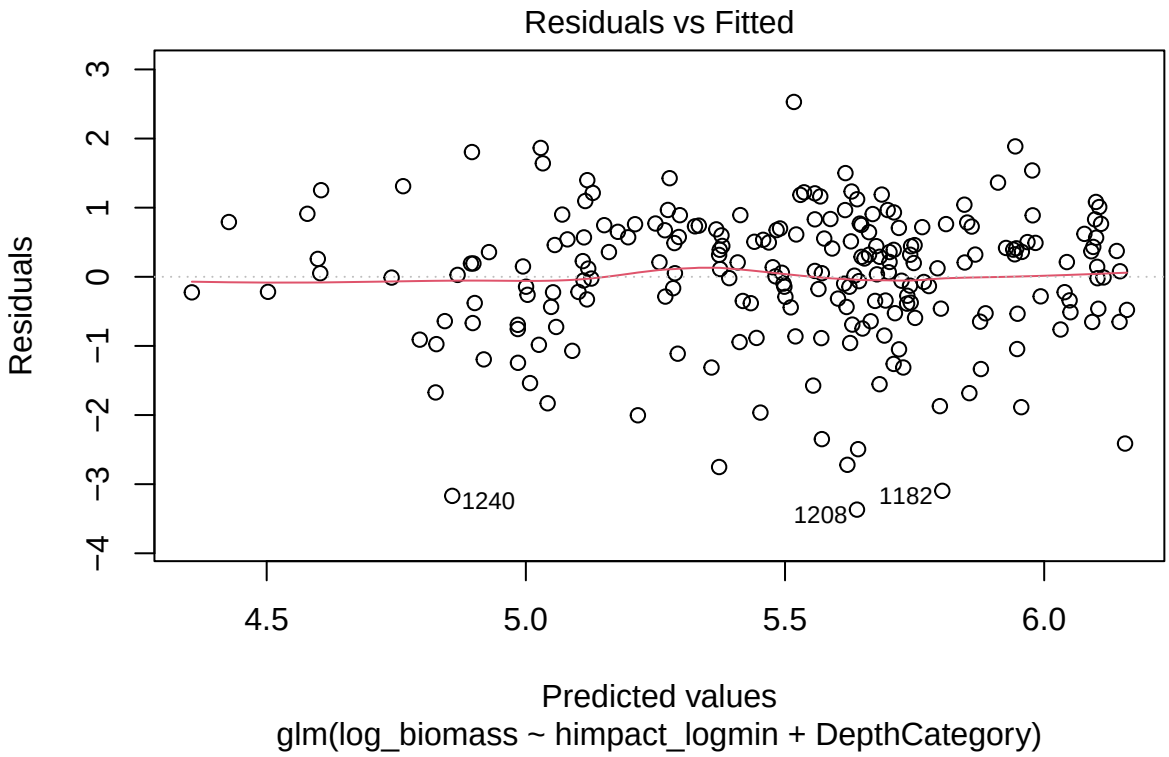
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

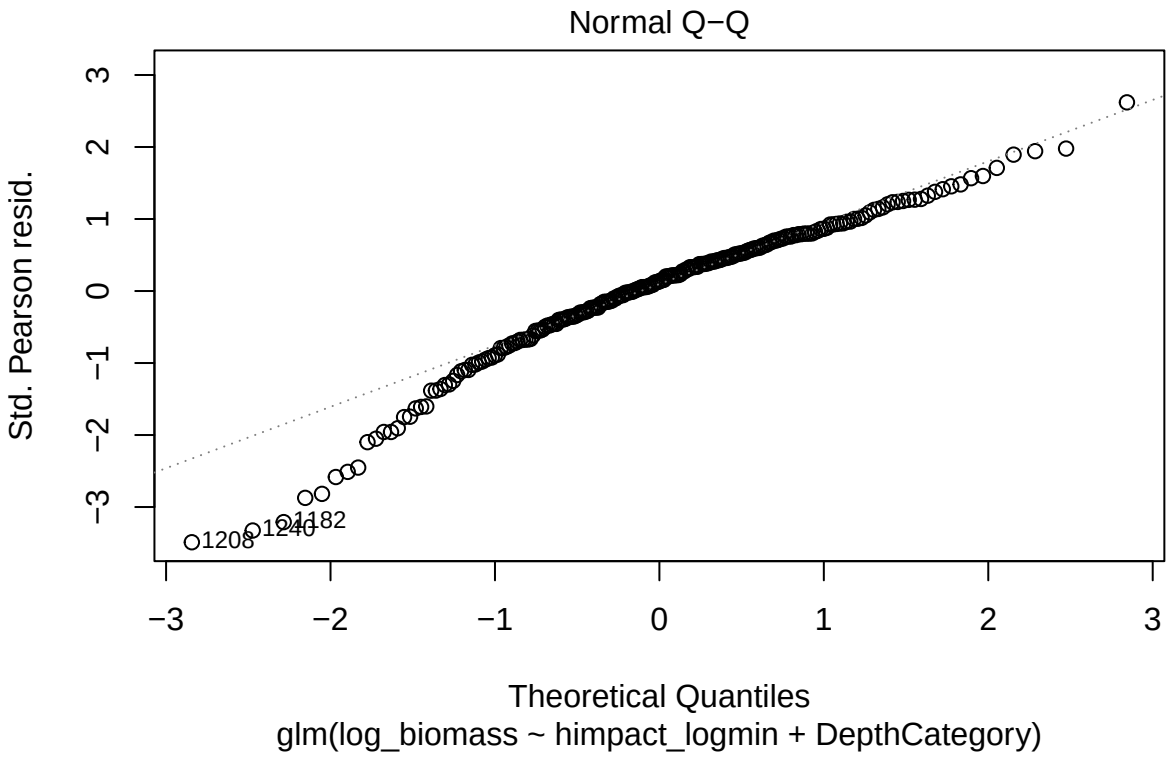


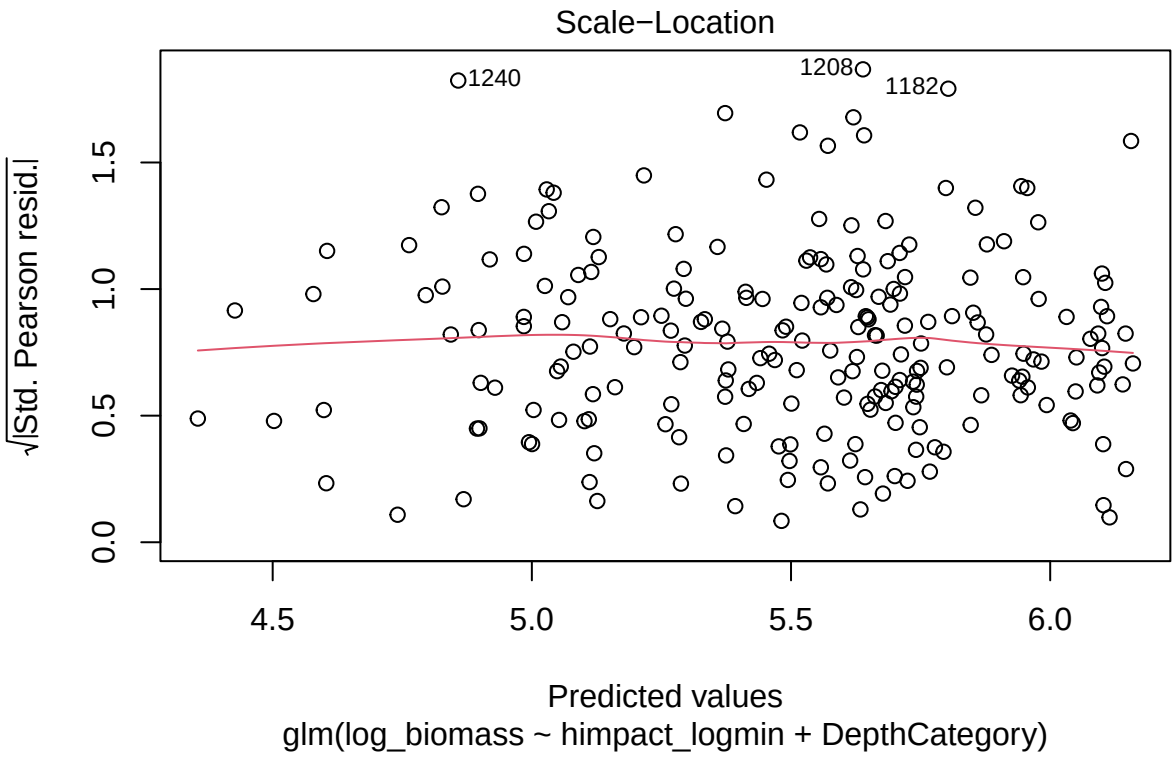

```
#plot residuals vs fittet
ggplot()+geom_point(aes(y=resid(glm_biomass_hab), x=fitted(glm_biomass_hab)))+ theme_classic()+ylab("Res
```

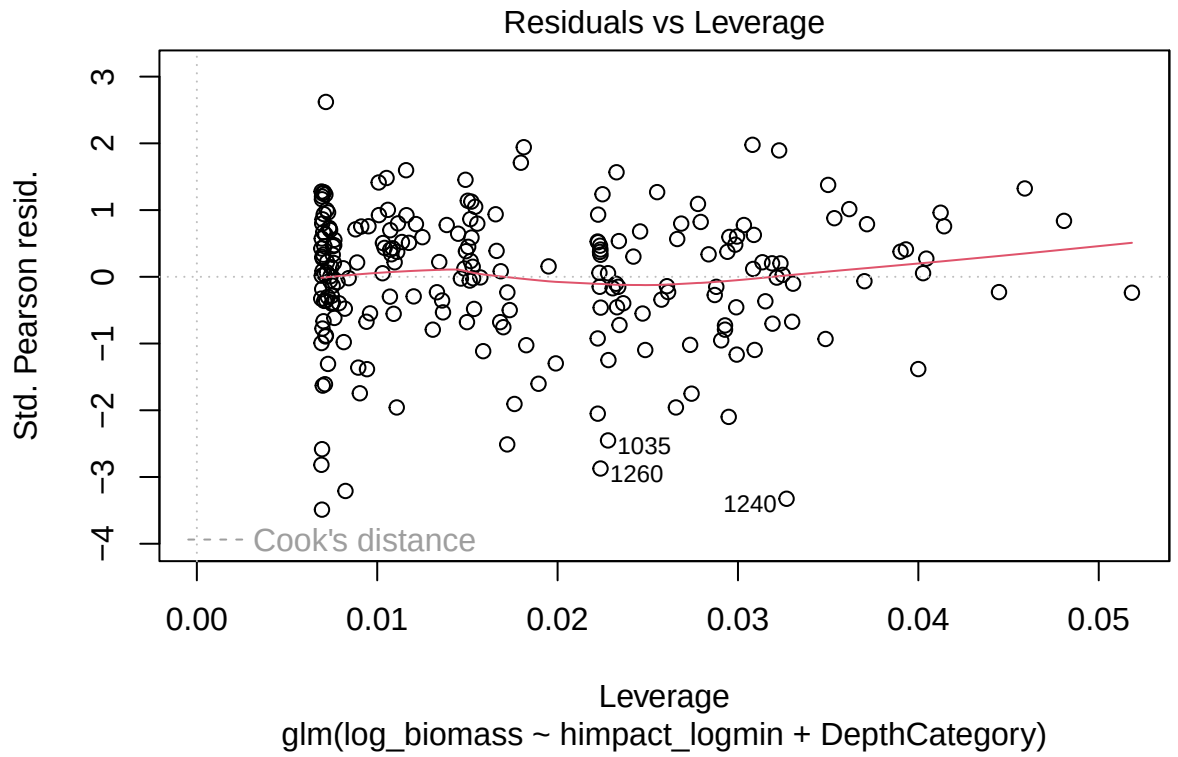



```
#plot fit diagnostics
plot(glm_biomass_hab)
```


Residuals vs Fitted
Residuals
3
2
1
0
−1
−2
−3
−4
1240
1208
1182
4.5
5.0
5.5
6.0
Predicted values
glm(log_biomass ~ himpact_logmin + DepthCategory)


Normal Q−Q
Std. Pearson resid.
−3
−2
−1
0
1
2
3
1208
1240
1182
Theoretical Quantiles
glm(log_biomass ~ himpact_logmin + DepthCategory)


Scale−Location
1240
1208
1182
√|Std. Pearson resid.|
0.0
0.5
1.0
1.5
4.5
5.0
5.5
6.0
Predicted values
glm(log_biomass ~ himpact_logmin + DepthCategory)

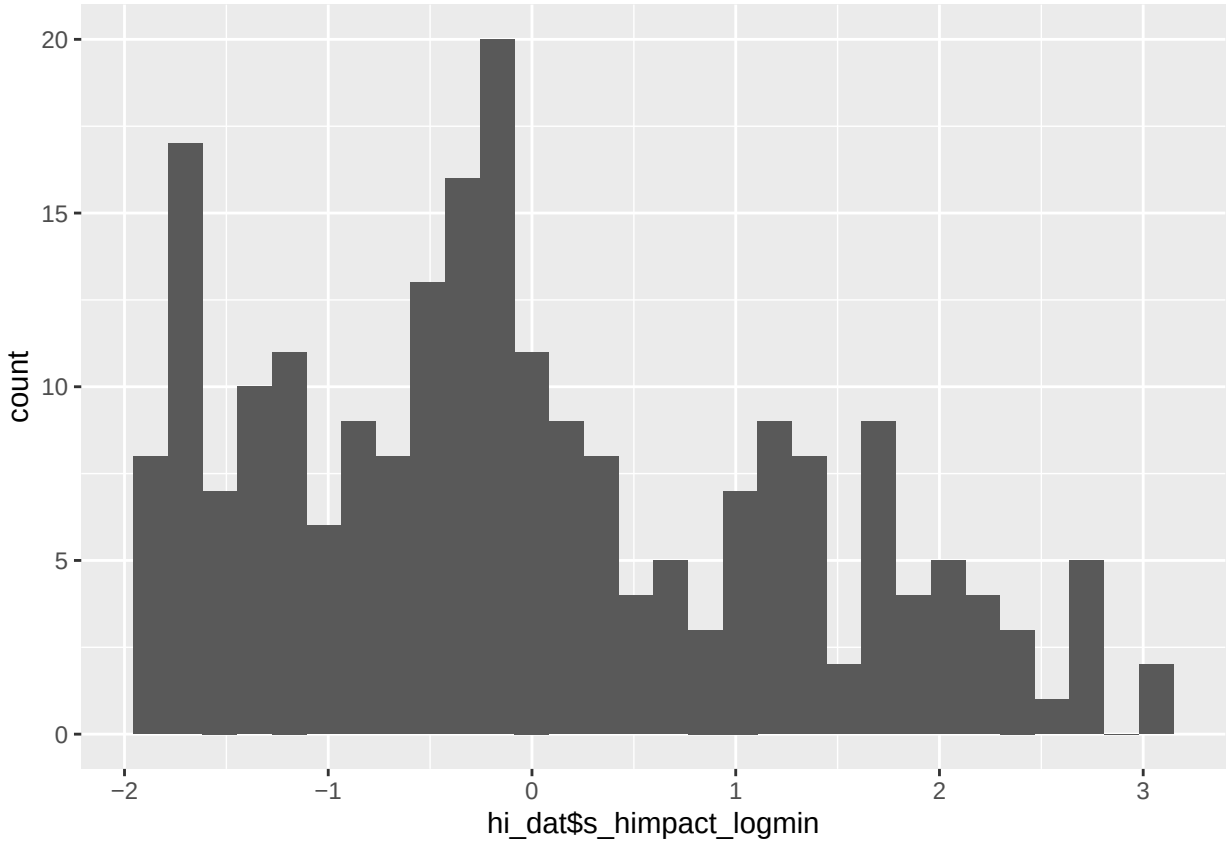


Often, when we use multiple regression, we may not be interested only on accounting form some sampling variance (e.g., depth), but instead we might want to understand which predictors are more strongly associated with our response variable. Here is when it becomes really important to (i) standardize our continuous predictors and (ii) relevel our categorical variables so the one with most data available is used as a reference.

You can't just choose the predictor with the largest estimated regression coefficient, because the predictors are all on different scales. For this, you need to center and scale the predictors. To make both continuous and categorical coefficients comparable, a good standardization for continuous variables is to subtract the mean and divide by two standard deviations. In other words:

```
#standardize human impact
hi_dat$s_himpact_logmin<-(hi_dat$himpact_logmin-mean(hi_dat$himpact_logmin))/2*sd(hi_dat$himpact_logmin
#plot standardized human impact
ggplot()+geom_histogram(aes(hi_dat$s_himpact_logmin))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Note the distribution stays the same, however now the human impact metric is scaled so that 0 is the mean log+min transformed human impact, and a score of 1 means that the score is two standard deviation about the mean.

Now refit the multiple regression suing the standardized human impact instead of the transformed human impact (as before).

```
#multiple regression with standardized data
glm_biomass_hab_s<-glm(log_biomass~s_himpact_logmin+DepthCategory,data=hi_dat, family="gaussian")
print(glm_biomass_hab_s)
```

```
##
## Call:  glm(formula = log_biomass ~ s_himpact_logmin + DepthCategory,
##     family = "gaussian", data = hi_dat)
##
## Coefficients:
##        (Intercept)    s_himpact_logmin   DepthCategory0-4m  DepthCategory4-10m
##             5.6050             -0.2895             -0.3753             -0.2118
##
## Degrees of Freedom: 223 Total (i.e. Null);  220 Residual
## Null Deviance:       241.4
## Residual Deviance: 206.5     AIC: 627.4
```

This tells us that depth being shallow has a relatively larger mean association than human impact.

Generalized linear models: Model selection

Many times in multiple regression we are also interested in knowing whether including some predictor or set of predictors increase or not the predictive accuracy of our model or not (i.e., if our model is better at predicting log-biomass if we include them). In other words, does our data support the inclusion of those variables as good predictors for biomass?

We could test this with model selection. I am not going to go in to the specific of how this is calculated, but a commonly used model selection tool is Aikake's information Criteria (AIC), that penalizes for the number of parameters estimated while accounting for the how good the model is in predicting our resonse variable. In R, this is computed with the AIC() function. Feel free to check the R documentation (?AIC) for more details in your spare time.

As we have fitted a model with human impact and one with human impact and depth category, we are going to test whether adding "Depth Category" improved or not our predictive capacity of predicting biomass (Note we use the standardized version because we used that in the simple linear model).

```
#model comparison
AIC(glm_biomass,glm_biomass_hab)
```

```
##                 df      AIC
## glm_biomass      3 628.3509
## glm_biomass_hab  5 627.4306
```

This returns the AIC value. What you need to know about this is that lower values indicate that the models are better in terms pf predicting our response variable, so in this case, it is telling us that including the depth variable, while increases the number of parameters our model has to estimate, it also increase our predictive capacity of log-biomass.

If you have extra time, on your own, feel free to recreate the above analyses (generalized linear models) for log_biomass with a different data subset (e.g., Australia).

Generalized linear models: logistic regression

For this section, we are going to do the same as above but using a different type of data generation process (presence/absence), assuming our data were generated from a binomial distribution.

To do so, we are going to use the hi_dat, but instead of biomass, we are going to use the "Scraping_potential".

- **`Scraping_potential`**: Potential scraping rates (area grazed per minute) for parrotfishes at each reef site were calculated as the product of parrotfish fish density, feeding rate, and bite dimension (area).

However, instead of using the raw metric, for this exercise we are going to transform the variable, and care about whether a given site had parrotfish or not (in other words if scraping potential is zero, it means that there were no parrotfish sampled at that site, and if the scraping potential is above zero, it means that parrotfish were recorded at that site). One example of why we would use only the presence/absence data, for example, is if we were interested in predicting the probability of observing parrotfish based on human impact.

So go ahead and create a new variable called "PA_parrot" that is "1" is scarping potential is above zero, and "0" is scraping potential is zero:

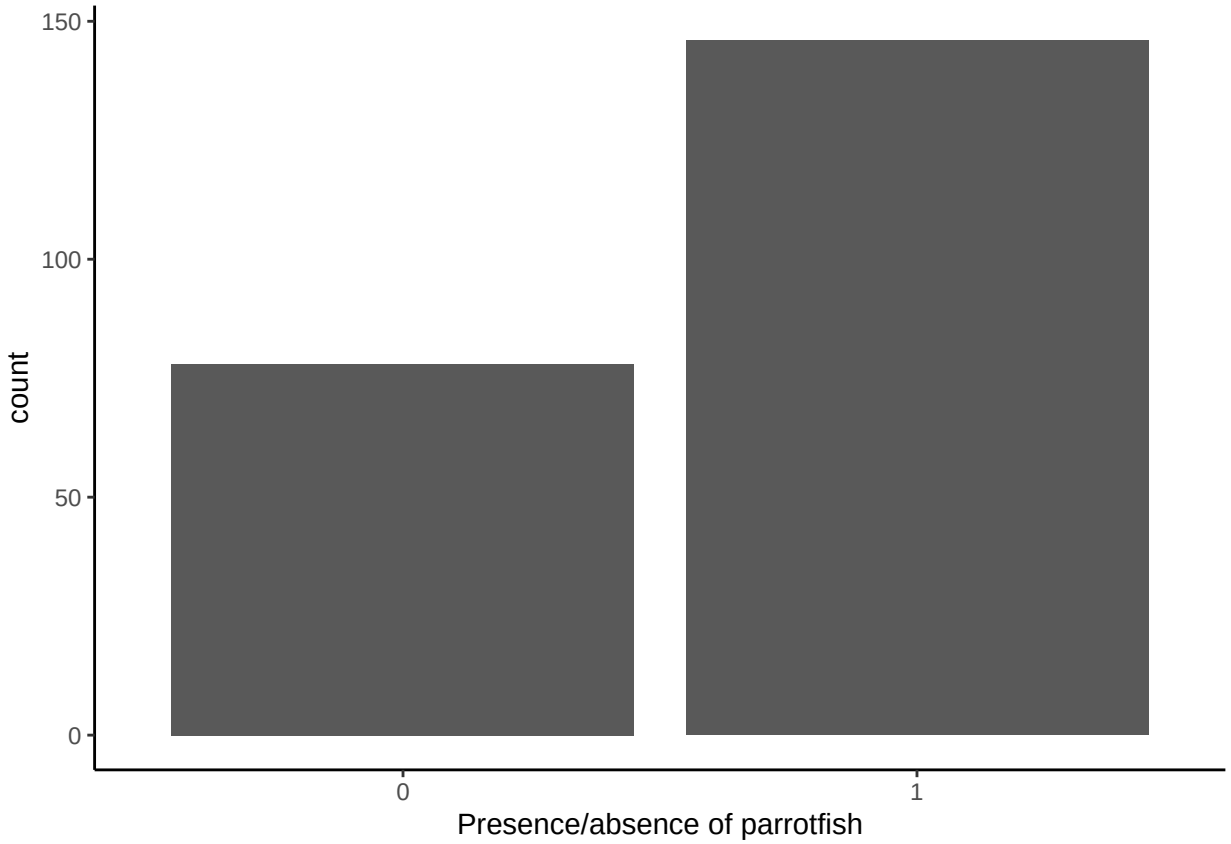
```
#summarize raw variable (to check for NAs)
summary(hi_dat$Scraping_potential)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00   15.93   33.40   44.54  433.75
```

```
#create new variable that indiactes wether parrotfish were present or not in a given site
hi_dat$PA_parrot<-as.factor(ifelse(hi_dat$Scraping_potential>0,1,0))
```

Visualize the distribution, with geom_bar():

```
#plot distribution
ggplot(hi_dat,aes(x=PA_parrot))+geom_bar()+theme_classic()+xlab("Presence/absence of parrotfish")
```



This data are not continuous (they are discrete) and do not follow a normal distribution. Thus, we would not use a "gaussian" family. Instead we would use a family distribution that resembles the data generation process of presence/absence data: binomial. The basic idea behind Generalized Linear Models is to specify a function that transforms the response space into a modeling space where we can perform a linear regression.

In logistic regression, we are modeling the relationship between the response and a set of predictors in log odds (log of the ratio of successes to failures) space.

The odds of success, i.e., the odds of observing parrotfish versus not observing them, is calculated as the number of sites parrotfish were sampled divided by the number of sites were not observed. The log odds if the log of the odds (i.e., logit), which is the sacle on which logistic regression is performed.

```
#calculate odds
observed_odds<-length(hi_dat$PA_parrot[hi_dat$PA_parrot=="1"])/length(hi_dat$PA_parrot[hi_dat$PA_parrot=
#calculate log odds
observed_logodds<-log(observed_odds)
```

The odds ranges from 0 to infinity, with one meaning the same number of sites observed parrotfish and didnt observed them. The log odds has some nice properties for linear modeling: It is symmetric around zero, positive log odds means that success is more likely than failure , and negative log odds means that failure is more likely than success.

Ok, now that we know the theoretical basics, what if we want to predict the probability of observing parrotfish from human impact accounting for depth? Well, in R, with the glm() function this is quite similar to what we have already done: we only have to change the response variable and the family distribution of our previous model called "glm_biomass_hab":

```
#binomial regression with PA_parrot
glm_parrot_hab<-glm(PA_parrot~himpact_logmin+DepthCategory,data=hi_dat, family="binomial")
print(glm_parrot_hab)
```

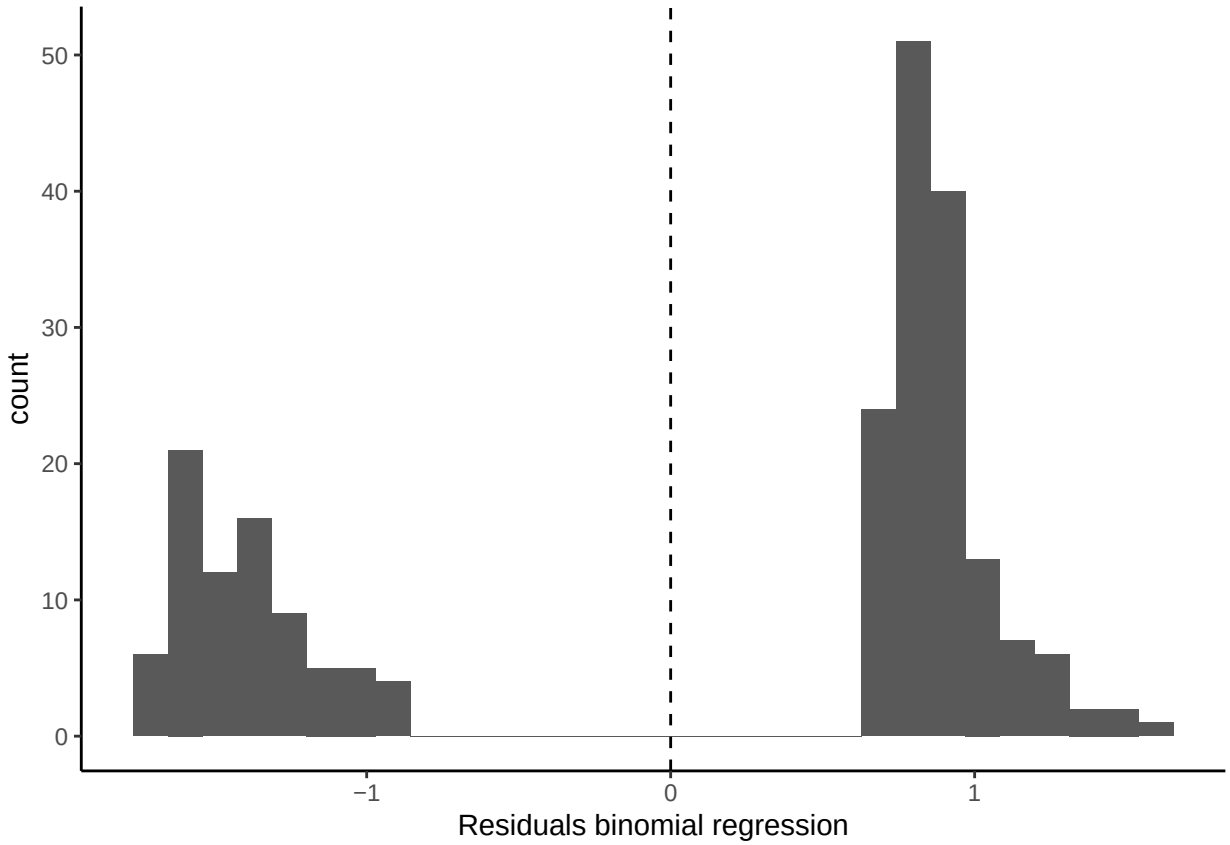
```
##
## Call:  glm(formula = PA_parrot ~ himpact_logmin + DepthCategory, family = "binomial",
##     data = hi_dat)
##
## Coefficients:
##        (Intercept)      himpact_logmin   DepthCategory0-4m  DepthCategory4-10m
##             1.5446             -0.1729             -1.0764             -0.3125
##
## Degrees of Freedom: 223 Total (i.e. Null);  220 Residual
## Null Deviance:       289.6
## Residual Deviance: 277.9     AIC: 285.9
```

As before, the intercept is the log-odds of observing parrotfish in the absence of human impact and for >10m depth. The other are the regression coefficients for transformed human impact and depth categories.

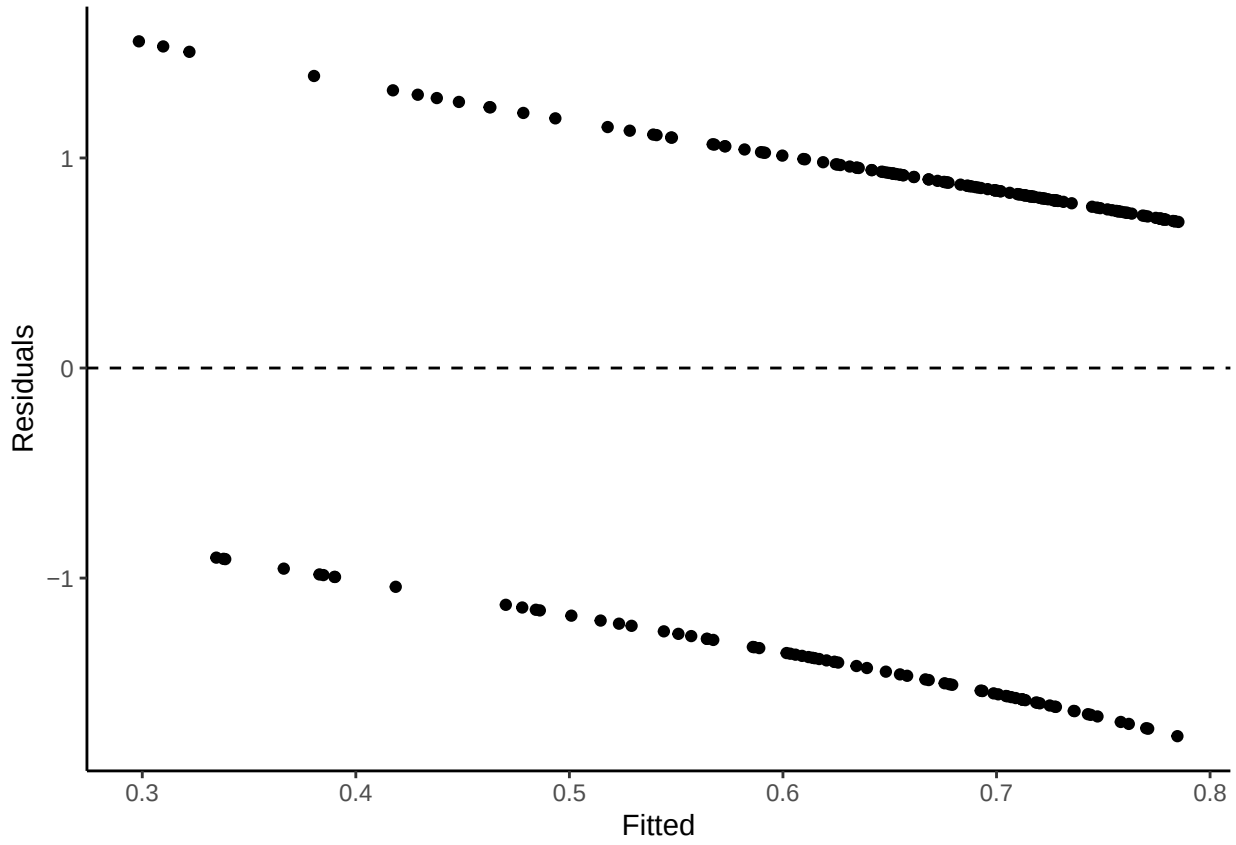
Before interpreting it, we have to check model fit. Note that now our random error structure is not normal (but instead has to have two modes).

```
#plot distribution of residuals
ggplot()+geom_histogram(aes(x=resid(glm_parrot_hab)))+ theme_classic()+xlab("Residuals binomial regressi
```

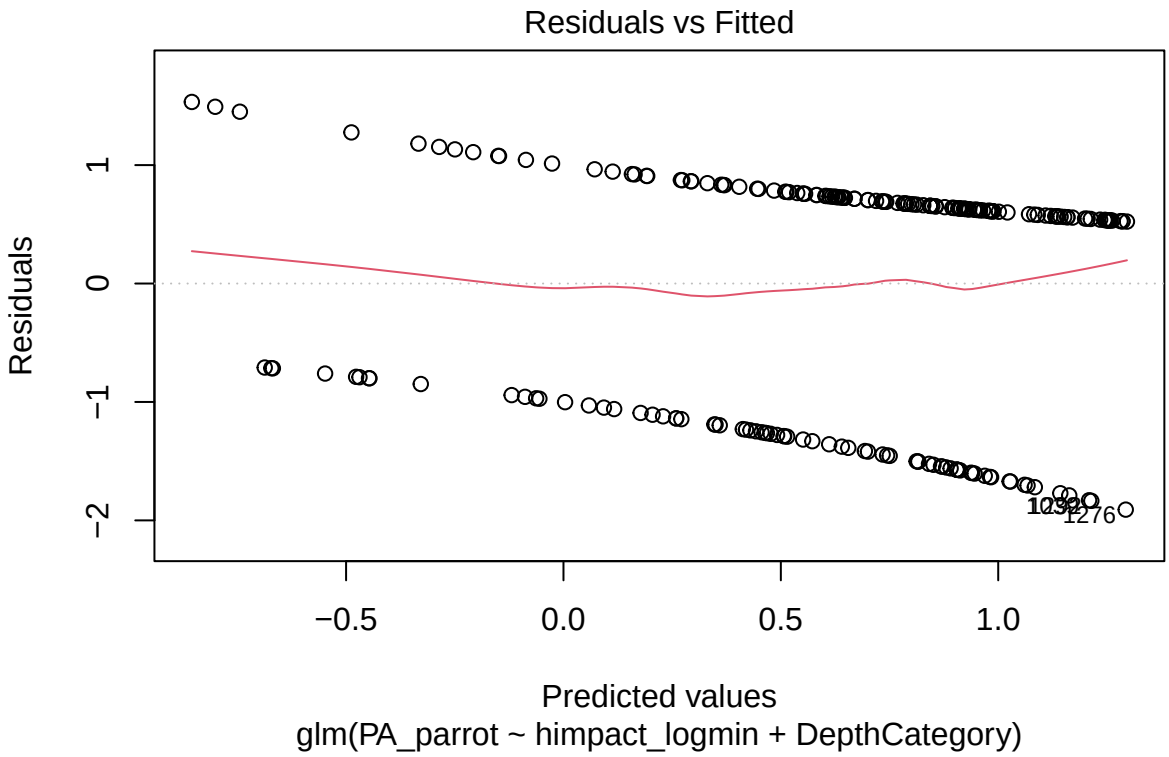
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

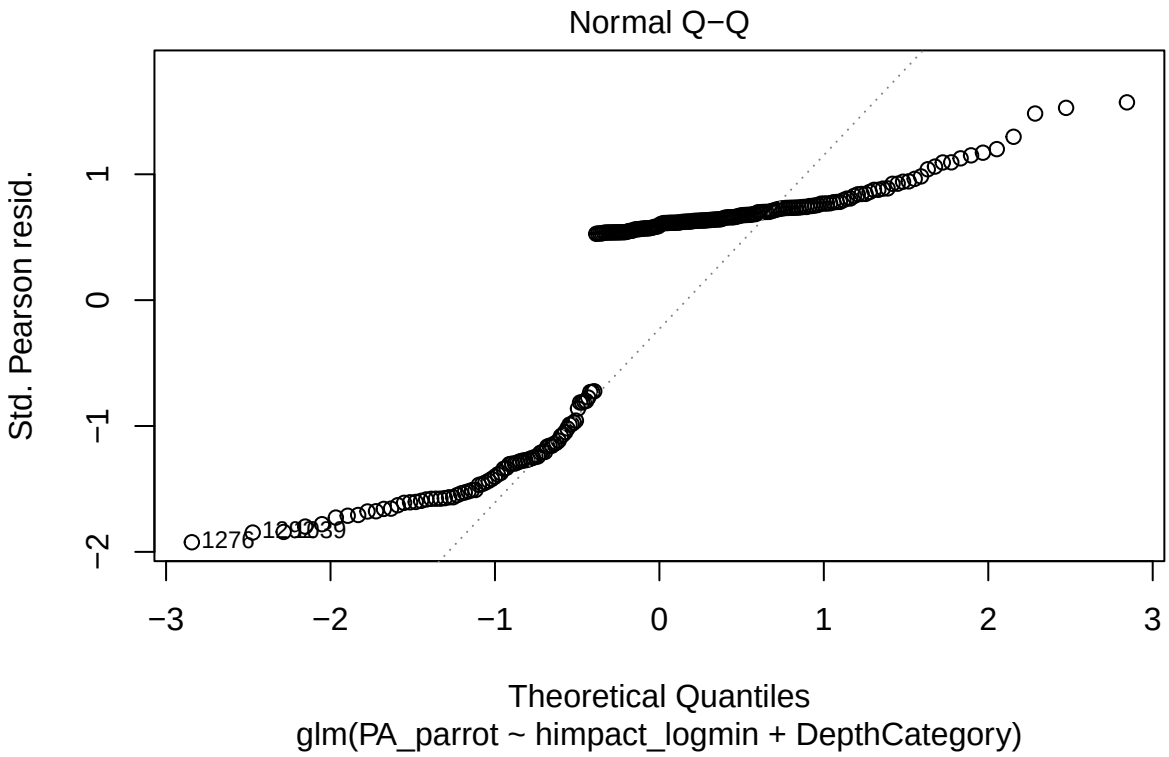


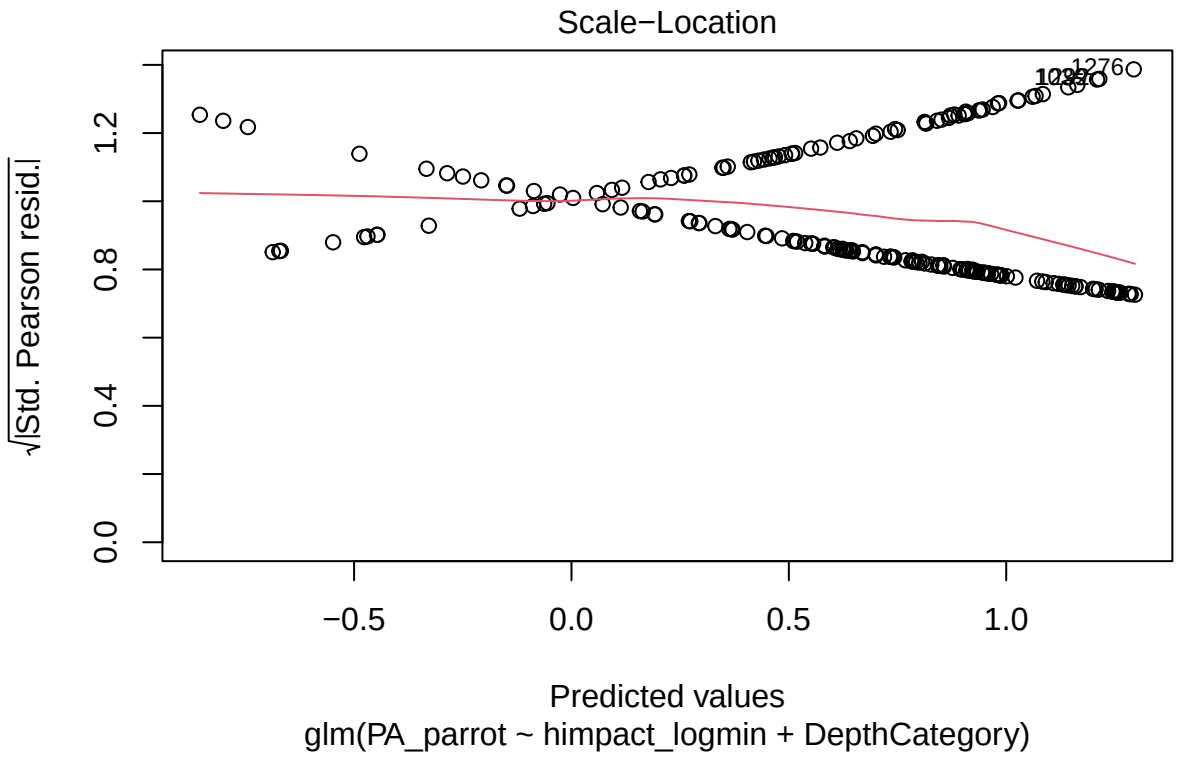

```
#plot residuals vs fittet
ggplot()+geom_point(aes(y=resid(glm_parrot_hab), x=fitted(glm_parrot_hab)))+ theme_classic()+ylab("Resi
```

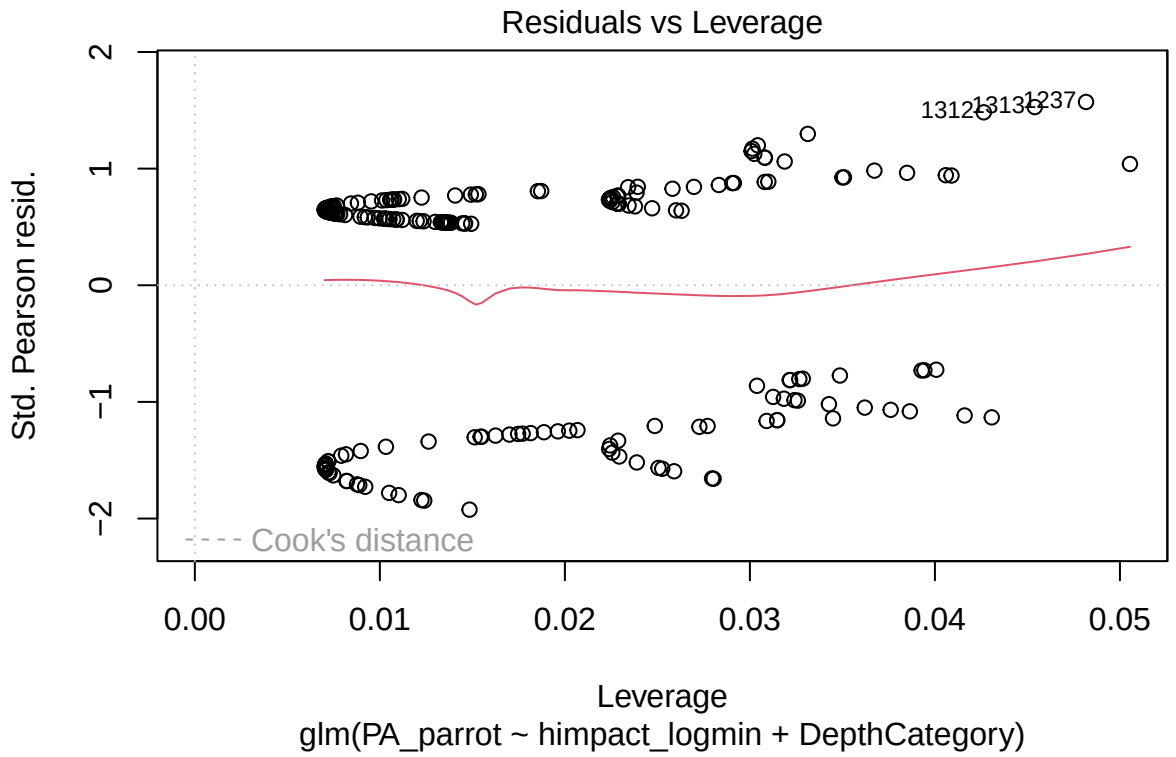



```
#plot fit diagnostics
plot(glm_parrot_hab)
```


Residuals vs Fitted
Residuals
1
0
−1
−2
−0.5
0.0
0.5
1.0
1276
Predicted values
glm(PA_parrot ~ himpact_logmin + DepthCategory)

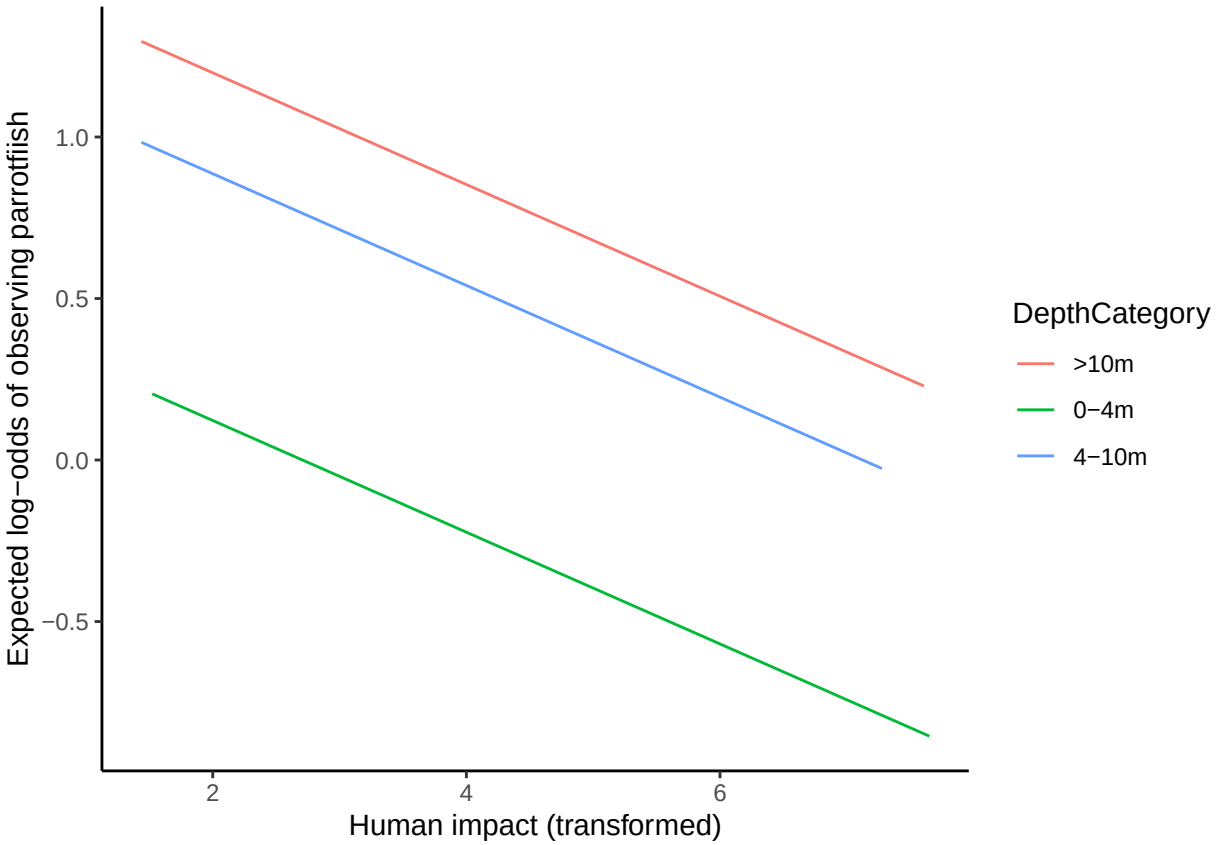

Normal Q−Q
Std. Pearson resid.
−2
−1
0
1
−3
−2
−1
0
1
2
3
1276
Theoretical Quantiles
glm(PA_parrot ~ himpact_logmin + DepthCategory)


Scale−Location
√|Std. Pearson resid.|
0.0
0.4
0.8
1.2
−0.5
0.0
0.5
1.0
1276
Predicted values
glm(PA_parrot ~ himpact_logmin + DepthCategory)



The data fit well. We can now predict how the log-odds of observing parrotfish is expected to change with human impact and depth and visualize it (just like we did with log-biomass):

```
#predict log-odds of observing parrotfish
predicted_logodds<-predict(glm_parrot_hab)
#plot log-odds of observing parrotfish given a sites human impact and depth
ggplot()+geom_line(data=hi_dat,aes(y=predicted_logodds,x=himpact_logmin,col=DepthCategory))+ylab("Expec
```

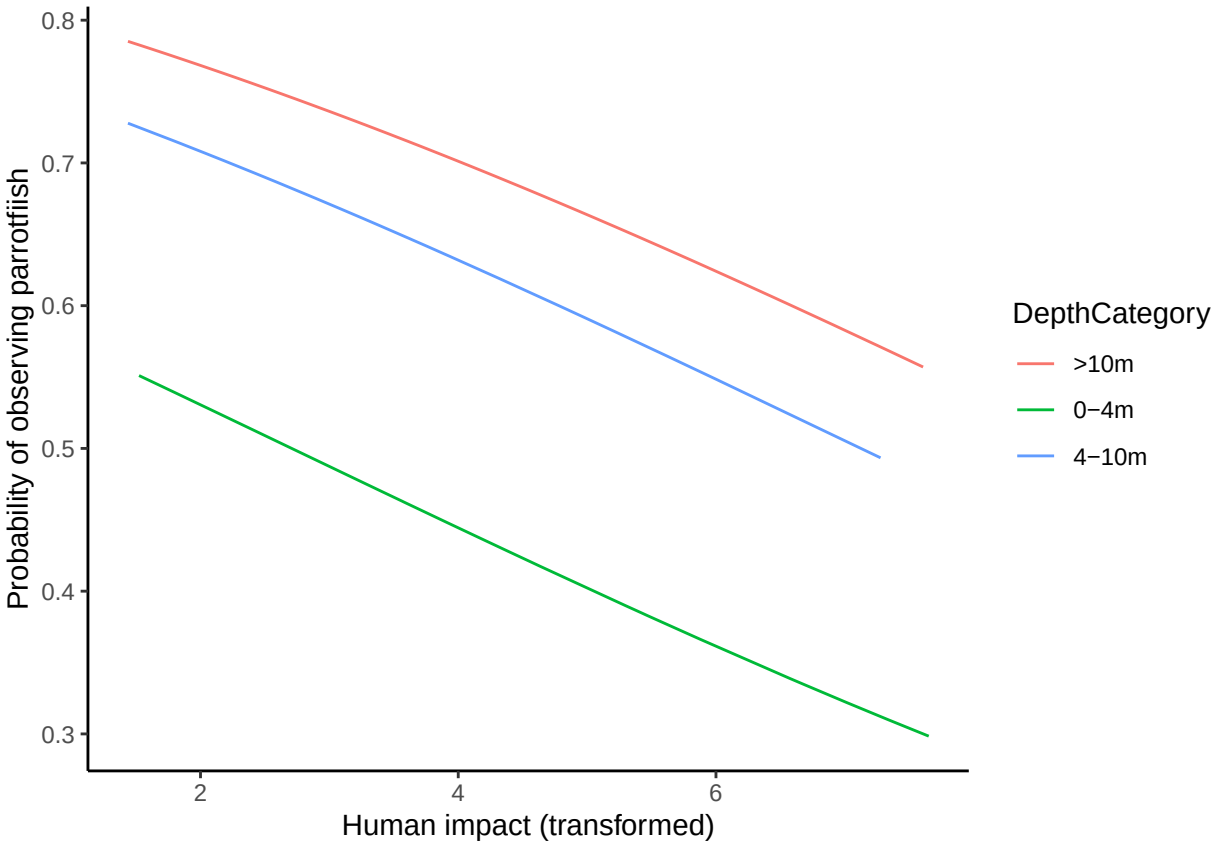


Our model suggests that the log-odds of observing parrotfish decreases with human impact and at shallower depth categories.

The log-odds is not very intuitive. However, from log-odds we can easily get to probabilities of observing parrotfish using the formula: "prob = exp(log-odds) / (1 + exp(log-odds))".

Go ahead and do the above plot but plotting probability instead of log-odds:

```
#predict probability of observing parrotfish
predicted_prob<-exp(predicted_logodds)/(1+exp(predicted_logodds))
#plot log-odds of observing parrotfish given a sites human impact and depth
ggplot()+geom_line(data=hi_dat,aes(y=predicted_prob,x=himpact_logmin,col=DepthCategory))+ylab("Probabil
```



This shows us how the probability of observing parrotfish is expected to decrease with human impact and depth category given our model.

Potential excercises:

1. Using the subset from New Caledonia, visualize how biomass of reef fish is predicted to change with human impact and depth, and interpret the results relative tho those obtained for Hawaii.
2. For the probability of observing parrotfish, perform model selection to check whether including habitat improves the predictive accuracy of our model (note you did this with biomass!).
3. Using the subset of Australia, visualize how the probability of observing top predators is predicted to change with human impact and depth, and interpret the results relative tho those obtained for Hawaii.
4. Theoretical: If you had count data (e.g., n_fish) instead of biomass, how would you modify the glm_biomass_hab model? (think through what family distribution you would use)

Generalized linear mixed effects models (OPTIONAL)

Many times our data is collected with nested or stratified sampling (e.g., repeated sites within different countries, repeated samples within different treatment categories..). Multilevel data is extremely common in marine science. To provide an example of how we might analyze this type of data in R, we are going to imagine we collected a subset of the reef_dat (i.e., that we only sampled reef sites from Hawaii, Seychelles and Madagascar). Go ahead and create a new dataset called "multilevel_dat" that filters only those countries (you did something similar today when you only got sites from Hawaii):

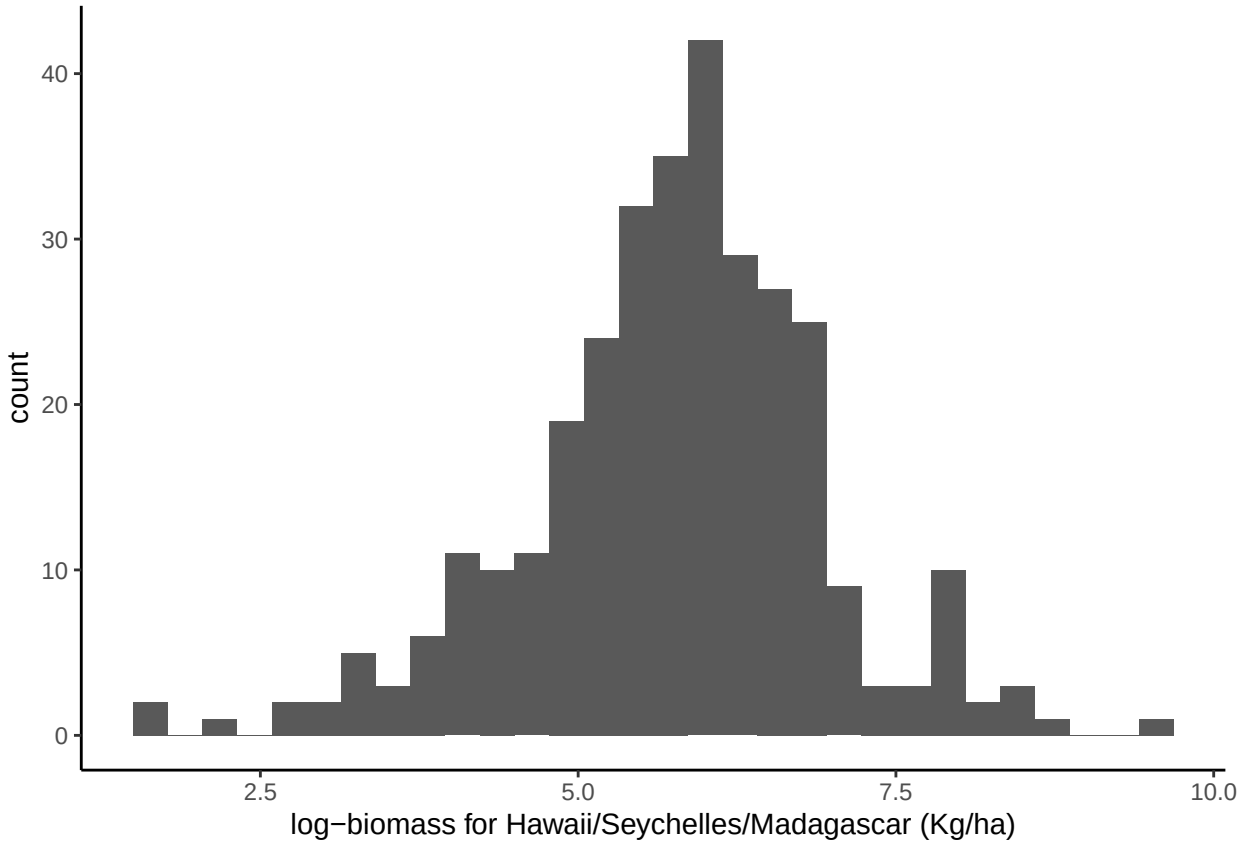
```
#subset reef dat
multilevel_dat<-droplevels(reef_dat[reef_dat$Larger=="Seychelles"|reef_dat$Larger=="Madagascar"|reef_da
```

You can do this with both log_biomass or the presence/absence of parrotfish. For this section, I will use log_biomass (gaussian). However, feel free to use the presence/absence of parrotfish also (binomial) if you have extra time.

Visualize the distribution of log_biomass for this subset of data.

```
#plot log-biomass distribution
ggplot(multilevel_dat,aes(x=log_biomass))+geom_histogram()+ theme_classic()+xlab("log-biomass for Hawai
```

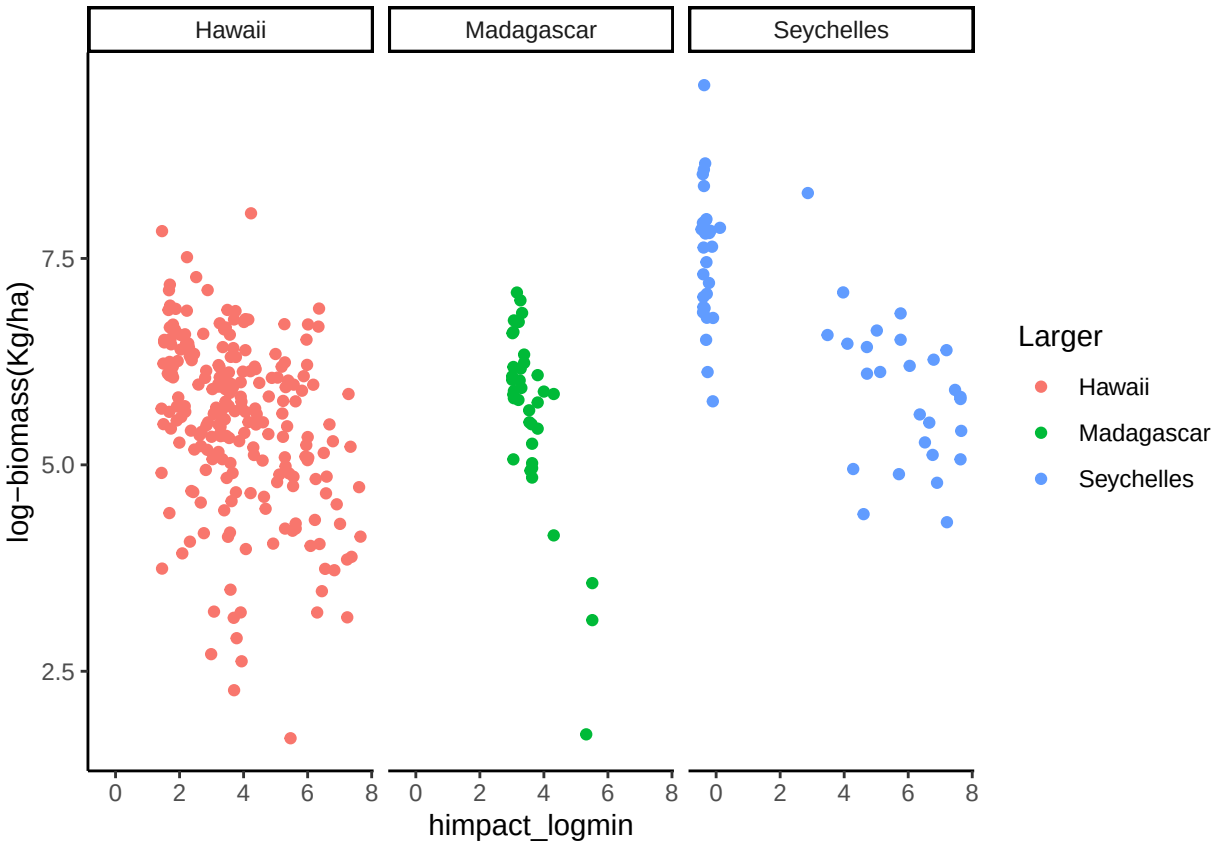
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Take a moment to think about the sampling design of the implied data. We want to predict biomass from human impact, accounting for depth. However, we sampled in different countries. Different countries, for example, could have different starting biomass points (i.e., intercept; for example if ecological capacity or productivity is different among countries).

First, lets visualize the relationship between log-biomass and transformed human impact in R with geom_point, using different colours for the different countries:

```
#plot observed relationship between log-biomass and human impact separated by country
ggplot(multilevel_dat)+geom_point(aes(x=himpact_logmin,y=log_biomass, color=Larger))+ theme_classic()+y
  facet_wrap(vars(Larger))
```



We can see that countries are likely to differ in their staring log-biomass intercept and their relation to human impact. should we fit the same line to all countries? a totally different line for each country? Or something somewhere in between? What you do will depend on your research question. However, for this exercise we are interested in knowing how human impact is associated with biomass (generalizable to the population), accounting for (i) depth and (ii) country (i.e., sites sampled at different depths and within different countries). Thus, we are going to include country as a random effect (critical for when we have a nested structure that is unbalanced or has missing data)!

Basically the population intercept and slope are informed by all the data, but countries are allowed to depart from populations in terms of the intercept, the slope or both). Here, for the shake of simplicity, we are going to assume that the relationship with human-impact does not avry by country but that the the starting point (i.e., intercept) does

To do this in R, we will use the glmer() function from the lme4 package-which has similar notation to the glm() function, but adding the random effects. Install and load the lme4 package. The fit the model with a random effect for the intercept:

```
#load lme4 package
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 4.2.2
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 4.2.2
```

```
#fit multilevel model with random intercept only (1|Larger) for country
multi_mod <- glmer(log_biomass~ himpact_logmin+ DepthCategory+ (1| Larger), data=multilevel_dat,family=
```

```
## Warning in glmer(log_biomass ~ himpact_logmin + DepthCategory + (1 | Larger), :
## calling glmer() with family=gaussian (identity link) as a shortcut to lmer() is
## deprecated; please call lmer() directly
```

```
print(multi_mod)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: log_biomass ~ himpact_logmin + DepthCategory + (1 | Larger)
##    Data: multilevel_dat
## REML criterion at convergence: 878.7333
## Random effects:
##  Groups   Name        Std.Dev.
##  Larger   (Intercept) 0.5051
##  Residual             0.9402
## Number of obs: 318, groups:  Larger, 3
## Fixed Effects:
##        (Intercept)      himpact_logmin   DepthCategory0-4m  DepthCategory4-10m
##             7.0072             -0.2657             -0.2666             -0.1747
```

We can see that, similar to our linear multiple regression, we still have the estimated intercept a regression coefficients (fixed effects) for human impact and depth categories. However, besides that we also have "Random effects". This section is telling us how the intercept varies among countries (it gives us the standard deviation). If we want to extract the random effects we use ranef() and if we want to extract the fixed effects we use fixef(), using the model as an input variable. Go ahead and extract the random effects and fixed effects of our multilevel model:

```
#extract random effects
ranef(multi_mod)
```

```
## $Larger
##            (Intercept)
## Hawaii      -0.3973026
## Madagascar  -0.1462654
## Seychelles   0.5435680
##
## with conditional variances for "Larger"
```

```
#extract fixed effects
fixef(multi_mod)
```

```
##        (Intercept)     himpact_logmin  DepthCategory0-4m DepthCategory4-10m
##          7.0072299         -0.2656893         -0.2666236         -0.1746628
```

The random effects are telling us how the intercept of each country varies in comparison to the global intercept (that one given to you by the fixed effects). In otehr words, it is telling you that Seychelles is expected to have more biomass at the intercept conditions (0 human impact and >10m). ~7+0.54 log-biomass, whereas Hawaii is expected to start at a lower log-biomass point (i.e., ~7-0.4 log-biomass).

Excercise (OPTIONAL)

1. Check the model fit of the multilevel model (similar to what you did for "glm_biomass_hab").
2. Predict and visualize log-biomass from the multilevel model (similar to what you did for "glm_biomass_hab").
3. Perform a multilevel model with the PA_parrot response variable and the binomial distribution.

Mapping reef sites globally (OPTIONAL)

Let's end these exercises by creating a map and plotting the sites from the Cinner et al paper. This will help us understand the spatial extent of our reef sites. To do this we will use the libraries "rworldmap" and "ggplot2" (which is already within the tidyverse package). Install the "rworldmap" package and load the library to your script.

```
#Install package
#install.packages("rworldmap") #uncomment if you have not installed it yet
#load libraries
library(rworldmap)
```

```
## Loading required package: sp
```

```
## ### Welcome to rworldmap ###
```

```
## For a short introduction type :   vignette('rworldmap')
```

We will use the "getMap()" function to load a map with high resolution (we will call this object "newmap"), and tell R to return you the object class:

```
#get Map
newmap <- getMap(resolution = "high")
#class of object
class(newmap)
```

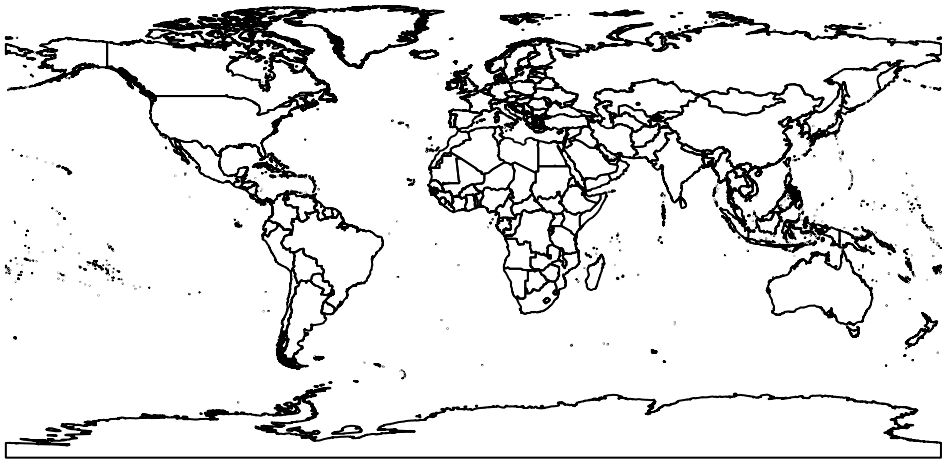
```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

You will see that a new object called "newmap" has appeared in your environment, and it is a Spatial Polygon Dataframe. This is basically a big data frame composed of polygons defined by a set of coordinates (e.g., latitude and longitude values).

We can plot this using the basic "plot()" function. However, as you have already learned, a more user-friendly way to modify plots as we want them is using the ggplot2 package. Check out the basic plot() function first:

```
#plot Map
plot(newmap)
```

```
## Warning in wkt(obj): CRS object has no comment
```



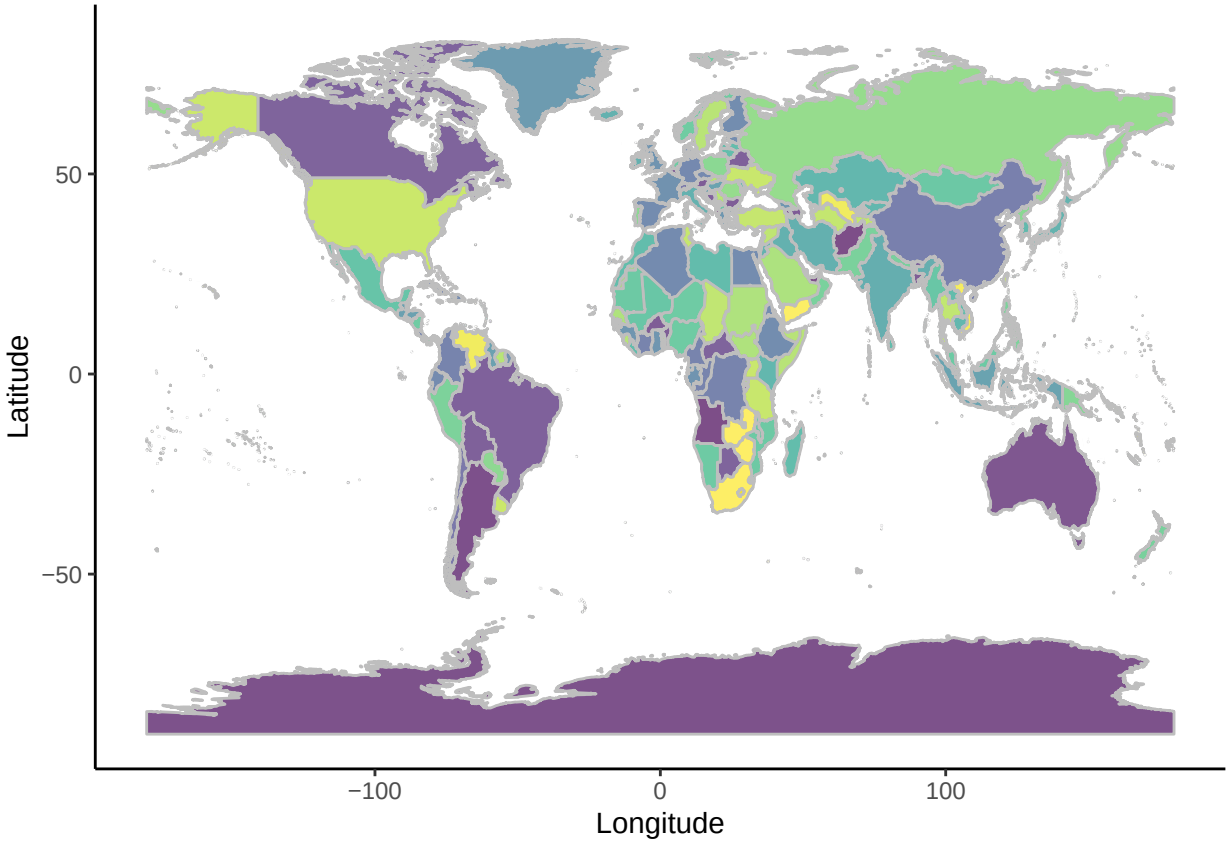
Now plot the map in ggplot. You will use the "geom_polygon()" function, use "long" as your x variable, "lat" as your y variable, and "group" as a grouping variable that indicates which coordinates to group together (country).

You can play around with ggplot's functions to make the plot how you want it. For example, you can use the "fill" command to colour by country (i.e., group), you can add labels to your axes, you an change the theme to add white (e.g., classic) background, you can also make colours slightly transparent (with the "alpha" value):

```
#plot map with ggplot
ggplot()+geom_polygon(data = newmap, aes(x=long, y = lat, group = group,fill = group), color = "grey", a
```

```
## Regions defined for each Polygons
```

```
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
## of ggplot2 3.3.4.
```

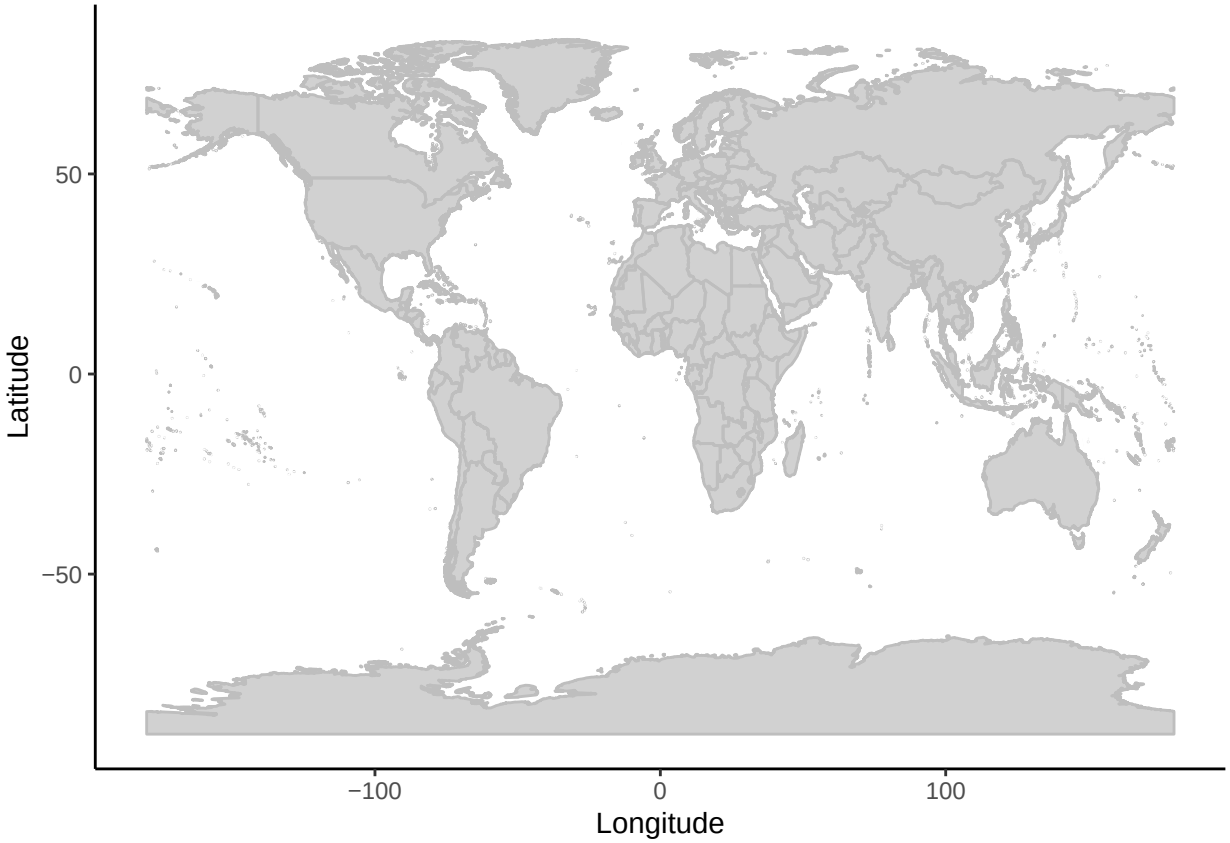


You can also create more simple maps. For example, create a simple grey map with a white classic background and axes labels. Assign the plot to an object called "map_reefsites":

```
#create a grey map object
map_reefsites<-ggplot()+geom_polygon(data = newmap, aes(x=long, y = lat, group = group), fill = "grey",
```

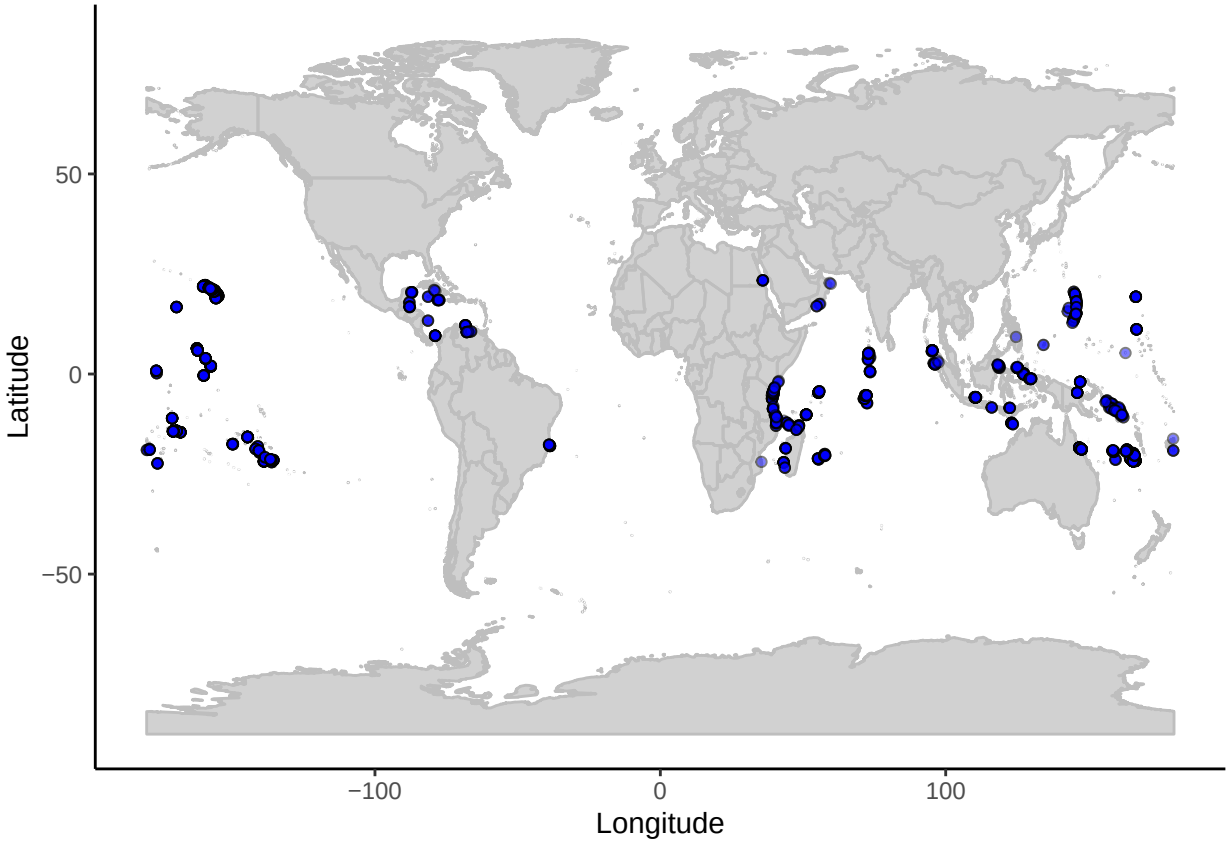
```
## Regions defined for each Polygons
```

```
#view map
map_reefsites
```



Now let's plot the reef sites on top. We do this by adding information to the already created map. Use the "Site_lat" and "Site_Long" variables.

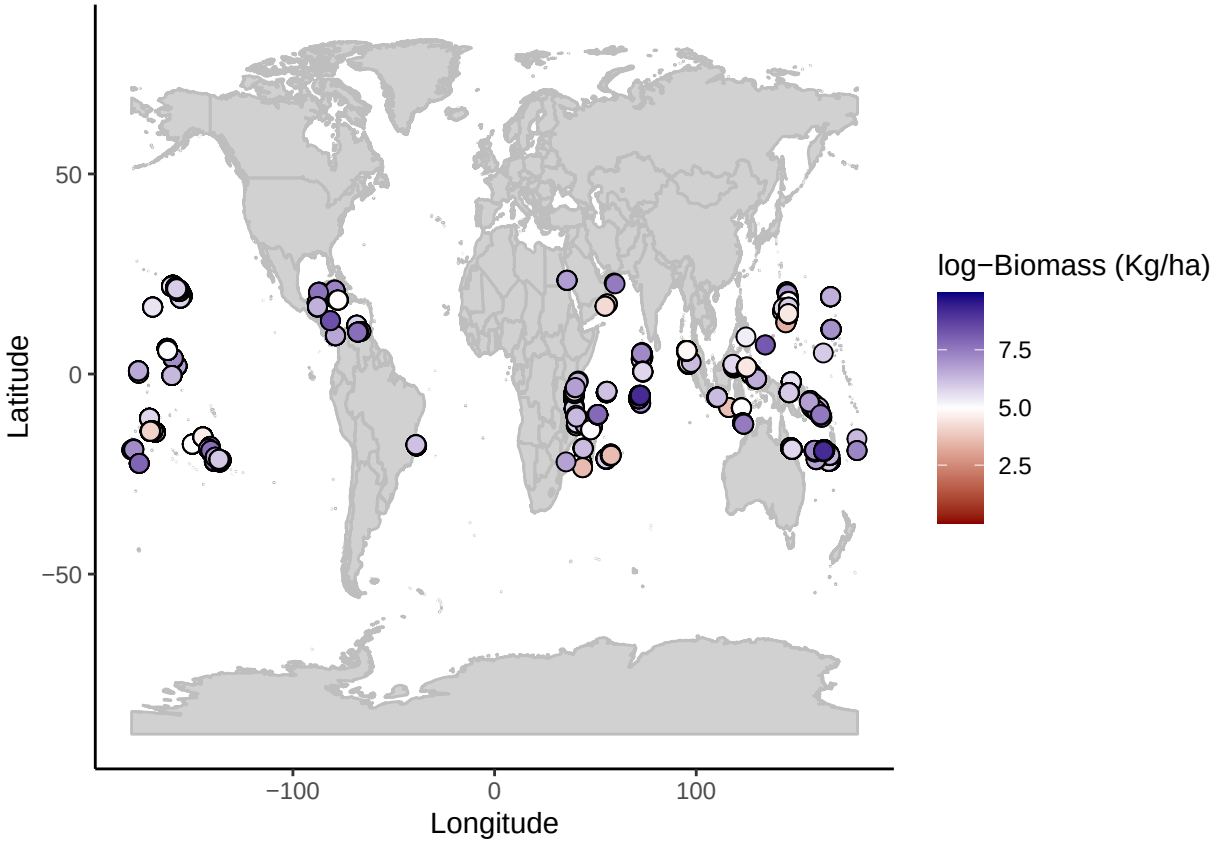
```
# map sites
map_reefsites+  geom_point(data=reef_dat,aes(x=Site_Long, y=Site_Lat),fill="blue",pch=21,alpha=0.5)
```



Note that I plotted the sites, and filled them with "blue" colour. The "pch" variable specifies which type of symbol we want. "21" is a point symbol with filling. In your own time, if you want to search for additional symbols look at the documentation that arises when you put ?pch. This will have a range of commands to visualize your symbols in different ways. At the bottom, under "'pch' values", you will be able to see different symbol specifications.

Go ahead and explore different plotting options. For example, you could "fill" your sites by the reef fish biomass they have:

```
map_reefsites+  geom_point(data=reef_dat,aes(x=Site_Long, y=Site_Lat,fill=log_biomass),pch=21,size=3)+s
```



Well done! Now we have a sense of where the 1798 sites are. You created a map with R and plotted the data in several ways!

We will leave you here so you use R for anything your imagination wants. Remember, in R it is all about giving it a go!

Now you know the basics of how to use this software. You can now use it to explore and analyze any type of data! Congratulations!

Please keep exploring! Together we can increase our knowledge on how to manage marine ecosystems in a way that we conserve them and also help them keep providing the key services they provide to human societies!