03\_model-analysis\_infection-endpoint\_load

Bryony Allen

09/05/2019

# Part 1: Import data

**N.B.** At the moment this code chuck reads in a .csv file with endpoint infection data, originally compiled in excel and then cleaned in “02\_tidy\_data-qpcr”script. In the future you will read in the .csv file of merged qPCR outputs (created, checked and cleaned in “02\_tidy\_data-qpcr”script) and experiment metadata (checked and cleaned in “02\_tidy\_data-metadata”script).

## Observations: 321  
## Variables: 26  
## $ ID <fct> A1.1, A1.2, A1.3, A1.4, A1.5, A1.6, A1...  
## $ Species <fct> Bb, Bb, Bb, Bb, Bb, Bb, Bb, Bb, Bb, Bb...  
## $ ExperimentNo <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## $ Scenario <fct> Coinfection, Coinfection, Coinfection,...  
## $ Treatment <fct> Rv-Bd, Rv-Bd, Rv-Bd, Rv-Bd, Rv-Bd, Rv-...  
## $ Exposure.1 <fct> rv, rv, rv, rv, rv, rv, rv, rv, rv, rv...  
## $ Exposure.2 <fct> bd, bd, bd, bd, bd, bd, bd, bd, bd, bd...  
## $ endpoint.date <fct> 06/06/2018, 26/05/2018, 06/06/2018, 27...  
## $ endpoint.code <fct> EU, MORT, EU, MORT, EU, MORT, EU, MORT...  
## $ Bd.endpoint.status <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## $ Bd.endpoint.CTmean <dbl> 30.94989, 37.30877, 32.19301, 36.65141...  
## $ Bd.endpoint.Qmean <dbl> 23.6021347, 0.4950310, 11.0839987, 0.7...  
## $ Rv.MCPendpoint.status <int> 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0,...  
## $ Rv.MCPendpoint.CTmean <dbl> 0.00000, 24.75441, 0.00000, 22.90014, ...  
## $ Rv.MCPendpoint.Qmean <dbl> 0.000000e+00, 2.462007e+05, 0.000000e+...  
## $ Rv.EBF3Nendpoint.status <int> 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0,...  
## $ Rv.EBF3Nendpoint.CTmean <dbl> NA, 33.46544, NA, 31.07020, NA, 34.558...  
## $ Rv.EBF3Nendpoint.Qmean <dbl> NA, 357.7631, NA, 1496.9165, NA, 182.4...  
## $ EMA.YN <int> 1, 0, 1, 0, 1, 1, 1, NA, NA, NA, NA, N...  
## $ EMA.date <fct> , NA, , NA, , , , NA, NA, NA, NA, NA, ...  
## $ EMA.GE.EMA <dbl> 0.004041085, NA, 0.479822159, NA, 0.00...  
## $ EMA.GE.WS <dbl> 0.026650012, NA, 0.005265172, NA, 0.00...  
## $ Bd.endpoint.GE <dbl> 236.021347, 4.950310, 110.839987, 7.37...  
## $ viable.GE <dbl> 0.004041085, NA, 0.479822159, NA, 0.00...  
## $ dead.GE <dbl> 0.022608927, NA, -0.474556987, NA, -0....  
## $ Rv.endpoint.load <dbl> NA, 1.376334e+03, NA, 9.135792e+02, NA...

# Part 2: Visualise Bd endpoint infection load data

Here I only include Bd infection loads over the 0.1 GE threshold.

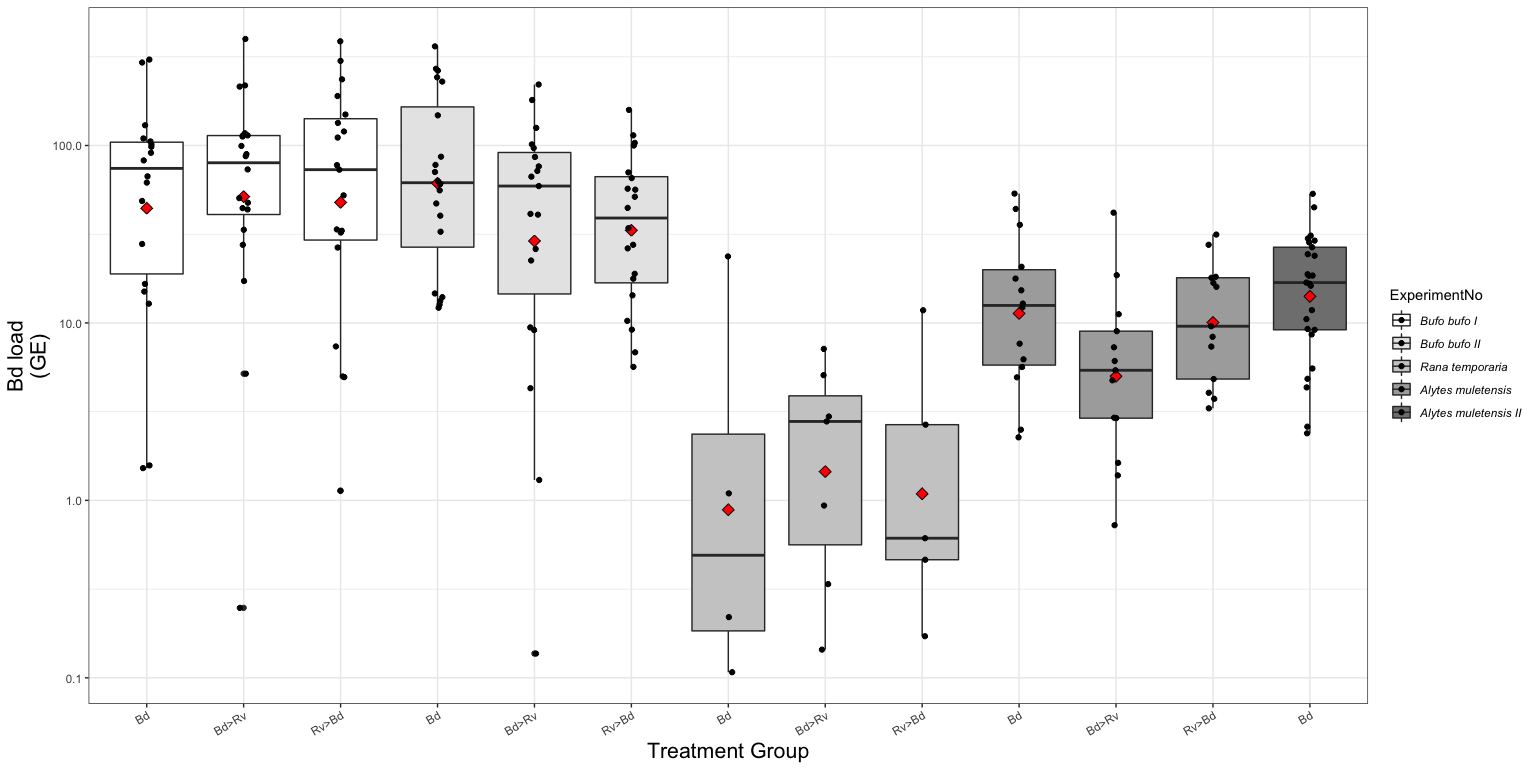


Fig.1. Boxplot of the endpoint Bd load for each treatment group across the three host species. Bd load is quantified as genomic equivalents (GE) where 1 GE represents 1 Bd zoospore. Where the black dots represent each sample and the red diamond the mean for that group.

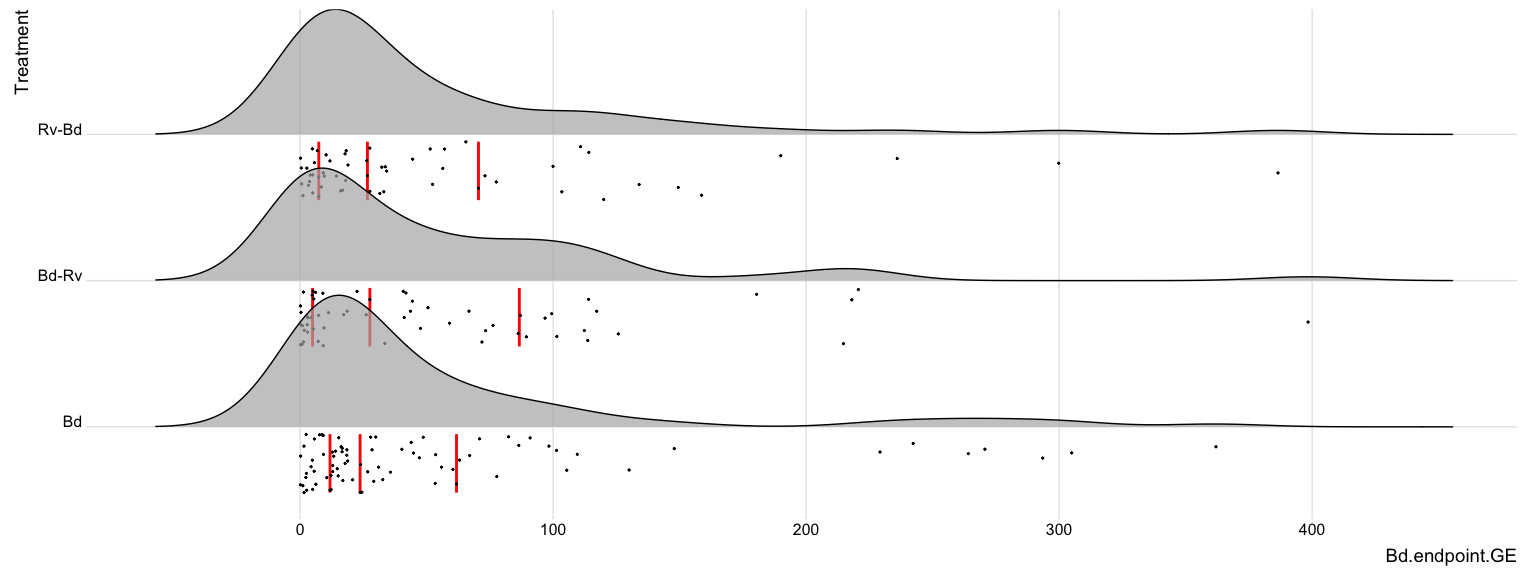


Fig. 2a. The distribution of Bd load expressed as GE from endpoint tissue samples.

… and when Bd load is logged

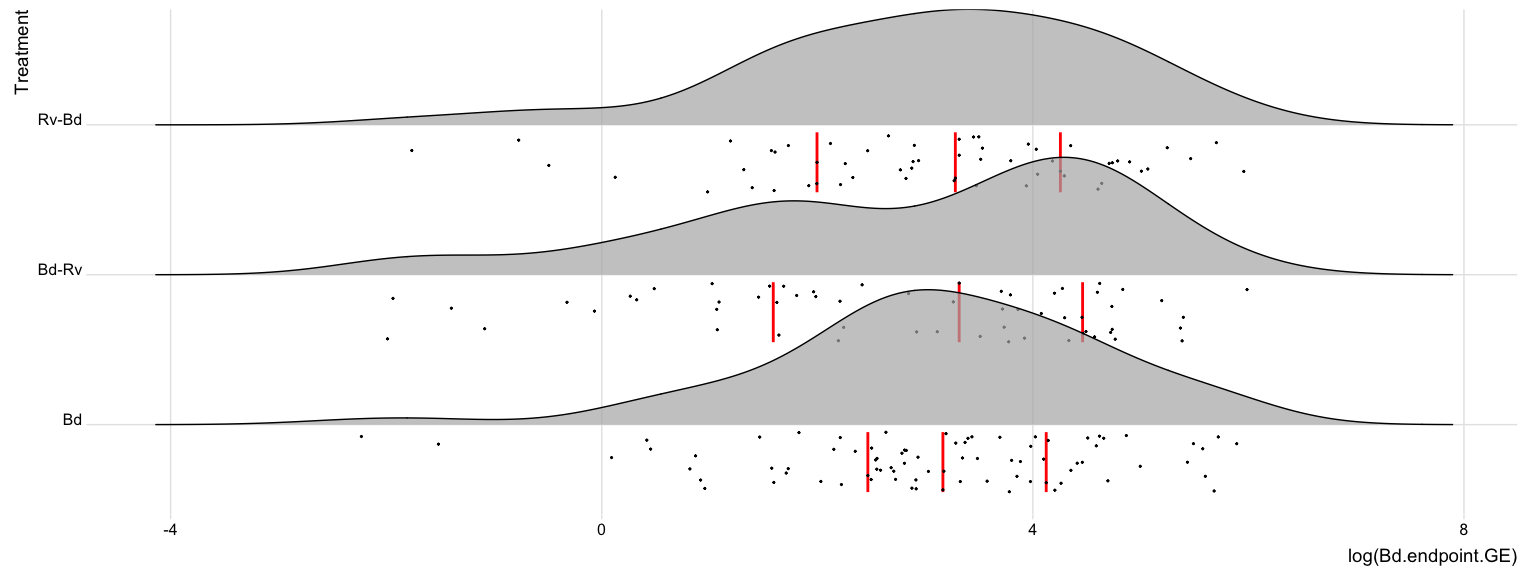


Fig. 2b. The logged distribution of Bd load expressed as genomic equivalents (GE), where one GE is equivalent to a single zoospore, from endpoint tissue samples.

# Part 3: Apply models to Bd Endpoint Infection Load

GLM’s where  
 response variable = endpoint load [continuous; ]  
 explanatory variable(s) = Treatment [categorical, levels = 4] & Species [categorical, levels = 5]

**N.B.** I use ExperimentNo as a proxy for species where

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Experiment No. | Species | total Bd zsp’s | min. temp. (oC) | max. temp. (oC) |
| 1 | *Bufo bufo* I | 3,675,000 | 16.6 | 23.5 |
| 2 | *Bufo bufo* II | 1,443,750 | 16.7 | 27.6 |
| 3 | *Rana temporaria* | 2,336,250 | 16.7 | 27.6 |
| 4 | *Alytes muletensis* I | 472,500 | 15 | 16.6 |
| 5 | *Alytes muletensis* II | 294,759 | 15 | 16.6 |

… as this also accounts for Bd dose and room temperature variation between experiments.

## Part 3a: Endpoint Infection Status: Bd

Here I create a dataframe with only individuals that am infected with Bd, at a level over the detection threshold of 0.1GE

?? **QUESTION** not sure whether I should be including the *Alytes muletensis* babies as they only have one treatment group (Bd only). Will this throw the models off?

Bd.load <- data.endpoint %>%  
 filter((Bd.endpoint.status=='1' & Bd.endpoint.GE > 0.1)) %>%   
 select(ID, Species, ExperimentNo, Scenario, Treatment, Bd.endpoint.status, Bd.endpoint.GE)   
  
droplevels(Bd.load)

### Model Selection:

Choosing a error structure family. I decided to do this with a semi-maximal model to help the fitting as much as possible.

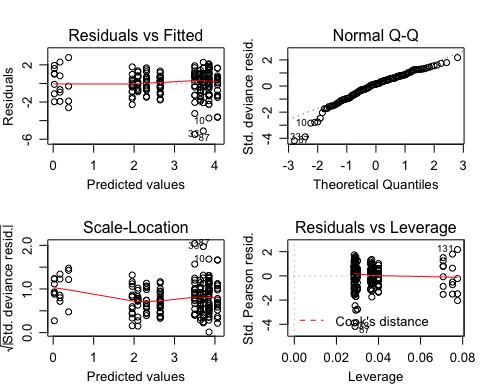
**N.B.** “the most appropriate link function is the one whcih produces the minimum residual deviance”

1. gaussian distribution with data log transformed

Bd.load2.1 <- glm(log(Bd.endpoint.GE) ~ Treatment + ExperimentNo, data=Bd.load, family= "gaussian")  
summary(Bd.load2.1) #AIC = 680.91 #Residual deviance: 337.15 on 190 degrees of freedom

##   
## Call:  
## glm(formula = log(Bd.endpoint.GE) ~ Treatment + ExperimentNo,   
## family = "gaussian", data = Bd.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.4879 -0.7098 0.2070 0.8337 2.7853   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.0639 0.2307 17.619 < 2e-16 \*\*\*  
## TreatmentBd-Rv -0.3526 0.2492 -1.415 0.159   
## TreatmentRv-Bd -0.2133 0.2507 -0.851 0.396   
## ExperimentNo2 -0.2110 0.2475 -0.852 0.395   
## ExperimentNo3 -3.6835 0.3774 -9.761 < 2e-16 \*\*\*  
## ExperimentNo4 -1.7573 0.2749 -6.393 1.23e-09 \*\*\*  
## ExperimentNo5 -1.4171 0.3524 -4.021 8.34e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 1.77445)  
##   
## Null deviance: 573.05 on 196 degrees of freedom  
## Residual deviance: 337.15 on 190 degrees of freedom  
## AIC: 680.91  
##   
## Number of Fisher Scoring iterations: 2

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Bd.load2.1)

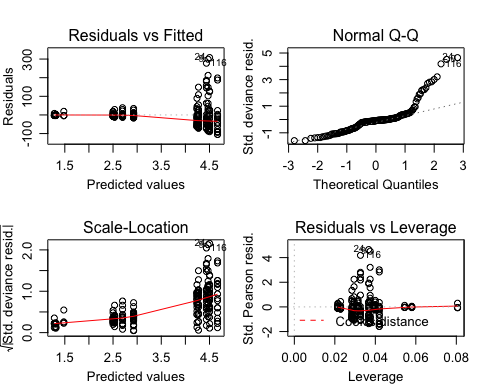
 - residuals vs fitted: not ideal, the scatter increases with fitted values  
- QQ plot is not great, (not normally distributed?)

1. gaussian distribution with log link

Bd.load2.2 <- glm(Bd.endpoint.GE ~ Treatment + ExperimentNo, data=Bd.load,family="gaussian"(link='log'))  
summary(Bd.load2.2) #AIC = 2227.3 #Residual deviance: 864687 on 190 degrees of freedom degrees of freedom

##   
## Call:  
## glm(formula = Bd.endpoint.GE ~ Treatment + ExperimentNo, family = gaussian(link = "log"),   
## data = Bd.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -104.832 -37.345 -5.296 8.066 308.193   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.6668 0.1299 35.917 <2e-16 \*\*\*  
## TreatmentBd-Rv -0.1647 0.1744 -0.944 0.3461   
## TreatmentRv-Bd -0.1962 0.1781 -1.102 0.2720   
## ExperimentNo2 -0.2304 0.1498 -1.538 0.1257   
## ExperimentNo3 -3.1876 4.3633 -0.731 0.4659   
## ExperimentNo4 -1.9695 0.8076 -2.439 0.0157 \*   
## ExperimentNo5 -1.7409 0.7350 -2.369 0.0189 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 4550.995)  
##   
## Null deviance: 1131056 on 196 degrees of freedom  
## Residual deviance: 864687 on 190 degrees of freedom  
## AIC: 2227.3  
##   
## Number of Fisher Scoring iterations: 6

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Bd.load2.2)

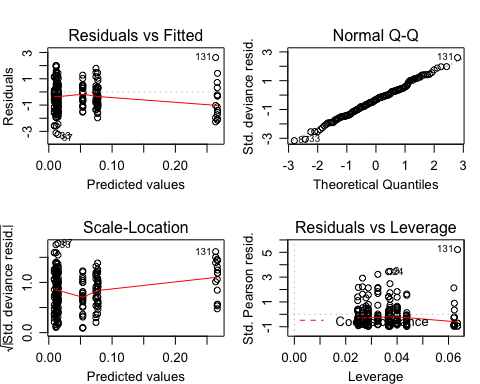
 Well this one is terrible … moving on …

1. Gamma distribution > This error structure is suitable for continous, positive, right skewed data where variance is near-constant on the log-scale

# Gamma distribution  
Bd.load2.3 <- glm(Bd.endpoint.GE ~ Treatment + ExperimentNo, data=Bd.load, family=Gamma())  
summary(Bd.load2.3) #AIC = 1832.7 #Residual deviance: 239.21 on 190 degrees of freedom

##   
## Call:  
## glm(formula = Bd.endpoint.GE ~ Treatment + ExperimentNo, family = Gamma(),   
## data = Bd.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.2336 -1.0492 -0.3168 0.2952 2.6140   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.008727 0.001895 4.606 7.50e-06 \*\*\*  
## TreatmentBd-Rv 0.002579 0.002671 0.965 0.335654   
## TreatmentRv-Bd 0.003009 0.002738 1.099 0.273168   
## ExperimentNo2 0.003125 0.002314 1.350 0.178521   
## ExperimentNo3 0.254890 0.068957 3.696 0.000286 \*\*\*  
## ExperimentNo4 0.065786 0.012603 5.220 4.67e-07 \*\*\*  
## ExperimentNo5 0.044893 0.011292 3.976 9.96e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for Gamma family taken to be 1.077433)  
##   
## Null deviance: 398.43 on 196 degrees of freedom  
## Residual deviance: 239.21 on 190 degrees of freedom  
## AIC: 1832.7  
##   
## Number of Fisher Scoring iterations: 7

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Bd.load2.3)

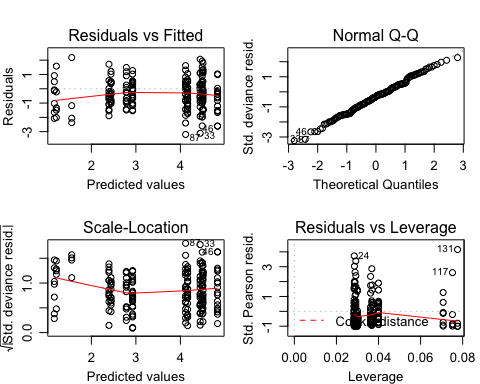
 The diagnostic plots are fine - residual vs. fitted do show patterning meaning the variance is non-consistent - QQ plot is really good - scale-location the scatter of the residuals decreases with the fitted values - residual vs. leverage  
- AIC is not good - residual deviance is better than log(Bd load) with gaussian

1. Gamma distribution with log-link > Apparent it is quite common to fit a log-link with a Gamma error structure … so that is left-skewed?

Bd.load2.4 <- glm(Bd.endpoint.GE ~ Treatment + ExperimentNo, data=Bd.load, family="Gamma"(link='log'))  
summary(Bd.load2.4) #AIC = 1829.2 #Residual deviance: 235.62 on 190 degrees of freedom

##   
## Call:  
## glm(formula = Bd.endpoint.GE ~ Treatment + ExperimentNo, family = Gamma(link = "log"),   
## data = Bd.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.1959 -1.0251 -0.3087 0.3372 2.1843   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.8351 0.1733 27.902 < 2e-16 \*\*\*  
## TreatmentBd-Rv -0.3904 0.1872 -2.085 0.0384 \*   
## TreatmentRv-Bd -0.3448 0.1884 -1.830 0.0688 .   
## ExperimentNo2 -0.3277 0.1859 -1.763 0.0796 .   
## ExperimentNo3 -3.2777 0.2835 -11.561 < 2e-16 \*\*\*  
## ExperimentNo4 -2.0499 0.2065 -9.927 < 2e-16 \*\*\*  
## ExperimentNo5 -1.9093 0.2647 -7.212 1.27e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for Gamma family taken to be 1.001522)  
##   
## Null deviance: 398.43 on 196 degrees of freedom  
## Residual deviance: 235.62 on 190 degrees of freedom  
## AIC: 1829.2  
##   
## Number of Fisher Scoring iterations: 7

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Bd.load2.4)



?? **QUESTION** log(Bd load) and gaussian or Gamma?

?? **QUESTION** can you compare AIC values of models that have difference error families? I thought it was used for assessing parameters contributions to the model.

### Model Comparison

The four models (all using log(Bd load) and gaussian error structure)

Bd.load1 <- glm(log(Bd.endpoint.GE) ~ Treatment \* ExperimentNo, data=Bd.load, family=gaussian)  
  
Bd.load2 <- glm(log(Bd.endpoint.GE) ~ Treatment + ExperimentNo, data=Bd.load, family=gaussian)  
  
Bd.load3 <- glm(log(Bd.endpoint.GE) ~ Treatment, data=Bd.load, family=gaussian)  
  
Bd.load4 <- glm(log(Bd.endpoint.GE) ~ ExperimentNo, data=Bd.load, family=gaussian)

?? **QUESTION** should I change the test = to better reflect the data type?

anova(Bd.load2, Bd.load4, test="Chisq") # compares Trt and Sp to just Sp

## Analysis of Deviance Table  
##   
## Model 1: log(Bd.endpoint.GE) ~ Treatment + ExperimentNo  
## Model 2: log(Bd.endpoint.GE) ~ ExperimentNo  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)  
## 1 190 337.15   
## 2 192 340.74 -2 -3.5936 0.3633

anova(Bd.load2, Bd.load3, test="Chisq") # compares Trt and Sp to just Trt

## Analysis of Deviance Table  
##   
## Model 1: log(Bd.endpoint.GE) ~ Treatment + ExperimentNo  
## Model 2: log(Bd.endpoint.GE) ~ Treatment  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)   
## 1 190 337.15   
## 2 194 570.32 -4 -233.18 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(Bd.load1, Bd.load2, test="Chisq") # compare more complex model to one with interaction

## Analysis of Deviance Table  
##   
## Model 1: log(Bd.endpoint.GE) ~ Treatment \* ExperimentNo  
## Model 2: log(Bd.endpoint.GE) ~ Treatment + ExperimentNo  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)  
## 1 184 328.44   
## 2 190 337.15 -6 -8.7081 0.5595

ANOVA 1: suggests we should reject the more complex model (Treatment + ExperimentNo) in favour for just the model with ExperimentNo only (pvalue = 0.3633)

ANOVA 2: suggests we should favour the more complex model (Treatment + ExperimentNo) over the model with Treatment only (pvalue = < .001) as adding ExperimentNo did lead to significantly improved fit ANOVA 3: shows adding a interaction term between Treatment and ExperimentNo did not significantly imporve fit (p-value = 0.5595)

**Conclusion**: we should choose the model with just ExperimentNo (aka species)

### Model Fit:

To see the fitted values from a regression object (the values of the dependent variable predicted by the model), access the fitted.values attribute from a regression object with `$fitted.values Look at <https://bookdown.org/ndphillips/YaRrr/linear-regression-with-lm.html>



Fig. 6. Plot of relationship between the fitted values from the model and the true data values.

**N.B.** Values should fall around the line but they seem to be clumped by ExperimentNo.

### Model Plotting:

To plot the model you need a range of values for which to produce fitted values. Then use the predict() function to create the model for all the values. predict() gives you the predicted values based on your (fitted) linear model, the argument type=“response” will give you the predicted probabilities

Bd.load4 <- glm(log(Bd.endpoint.GE) ~ ExperimentNo, data=Bd.load, family=gaussian)  
  
# create a dataframe of "new" data   
newdat <- expand.grid(ExperimentNo=c("1", "2", "3", "4", "5"),Treatment=c("Bd", "Bd-Rv", "Rv-Bd"))  
  
# predict the value/result of the new data using the glm  
newdat <-cbind(newdat, predict(object = Bd.load4, # the model   
 newdata=newdat, se=TRUE, type="response", print.matrix=T)) # dataframe of new data   
newdat

## ExperimentNo Treatment fit se.fit residual.scale  
## 1 1 Bd 3.869096 0.1764505 1.332172  
## 2 2 Bd 3.667092 0.1734340 1.332172  
## 3 3 Bd 0.159463 0.3330430 1.332172  
## 4 4 Bd 2.122707 0.2106349 1.332172  
## 5 5 Bd 2.646824 0.2664344 1.332172  
## 6 1 Bd-Rv 3.869096 0.1764505 1.332172  
## 7 2 Bd-Rv 3.667092 0.1734340 1.332172  
## 8 3 Bd-Rv 0.159463 0.3330430 1.332172  
## 9 4 Bd-Rv 2.122707 0.2106349 1.332172  
## 10 5 Bd-Rv 2.646824 0.2664344 1.332172  
## 11 1 Rv-Bd 3.869096 0.1764505 1.332172  
## 12 2 Rv-Bd 3.667092 0.1734340 1.332172  
## 13 3 Rv-Bd 0.159463 0.3330430 1.332172  
## 14 4 Rv-Bd 2.122707 0.2106349 1.332172  
## 15 5 Rv-Bd 2.646824 0.2664344 1.332172

expl.var <- c(1:3) # chose the range for the x-axis (Experiment No.)  
exp.labs <- c("1" = "Bufo bufo I", "2" = "Bufo bufo II", "3" = "Rana temporaria", "4" = "Alytes muletensis I", "5" = "Alytes muletensis II")  
  
newdat1<- subset(newdat, ExperimentNo== "1") # need to subset the data so you can plot each seperatly   
newdat2<- subset(newdat, ExperimentNo=="2")  
newdat3<- subset(newdat, ExperimentNo=="3")  
newdat4<- subset(newdat, ExperimentNo=="4")  
newdat5<- subset(newdat, ExperimentNo=="5")

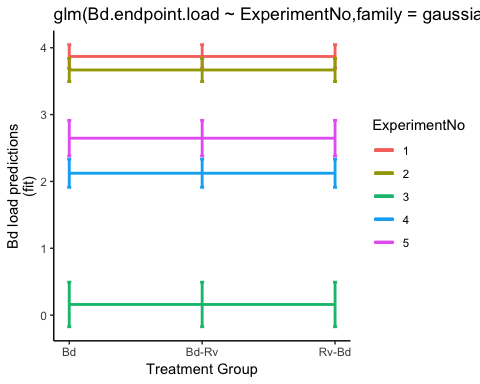


Fig. 7. Probability of Bd infection load (GE) predicted by model

Sorry I can’t get the legend to change!

The Bd load predictions are interesting… The two *Bufo bufo* experiments have the highest predicted loads, and reflect the Bd dosed (1 = 3,675,000 zsp’s; 2 = 1,443,750 zsp’s). The *Alytes muletensis* are also clumped, here it’s either a reflection of Bd dose (4 = 472,500 zsp’s; 5 = 294,759 zsp’s), which was scaled to container size, or development stage. If development stage, we see the larger, more developed tadpoles having low loads than the babies.

A reminder, just in case …

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Experiment No. | Species | total Bd zsp’s | min. temp. (oC) | max. temp. (oC) |
| 1 | *Bufo bufo* I | 3,675,000 | 16.6 | 23.5 |
| 2 | *Bufo bufo* II | 1,443,750 | 16.7 | 27.6 |
| 3 | *Rana temporaria* | 2,336,250 | 16.7 | 27.6 |
| 4 | *Alytes muletensis* I | 472,500 | 15 | 16.6 |
| 5 | *Alytes muletensis* II | 294,759 | 15 | 16.6 |

### Models Checks

Here I check the two best models.

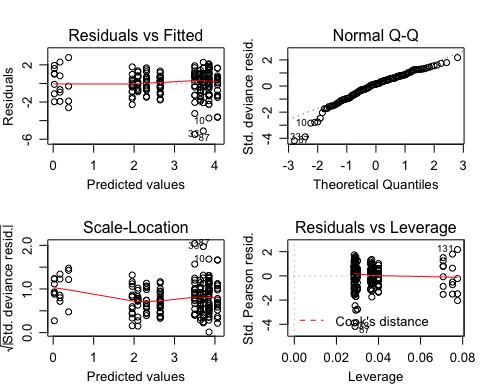
#### Maximal Model: Species + Treamtent (no interaction)

Fit the most complex model first, with the main effects for ExperimentNo. and Treatment.

Look at the estimates of the coefficients using summary()

##   
## Call:  
## glm(formula = log(Bd.endpoint.GE) ~ Treatment + ExperimentNo,   
## family = gaussian, data = Bd.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.4879 -0.7098 0.2070 0.8337 2.7853   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.0639 0.2307 17.619 < 2e-16 \*\*\*  
## TreatmentBd-Rv -0.3526 0.2492 -1.415 0.159   
## TreatmentRv-Bd -0.2133 0.2507 -0.851 0.396   
## ExperimentNo2 -0.2110 0.2475 -0.852 0.395   
## ExperimentNo3 -3.6835 0.3774 -9.761 < 2e-16 \*\*\*  
## ExperimentNo4 -1.7573 0.2749 -6.393 1.23e-09 \*\*\*  
## ExperimentNo5 -1.4171 0.3524 -4.021 8.34e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 1.77445)  
##   
## Null deviance: 573.05 on 196 degrees of freedom  
## Residual deviance: 337.15 on 190 degrees of freedom  
## AIC: 680.91  
##   
## Number of Fisher Scoring iterations: 2

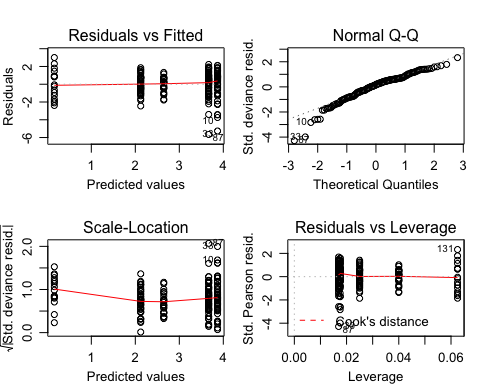
Residual deviance should be the same as the residual degrees of freedom but is way larger, indicating overdispersion (unexplained variation in the response).



#### Simplified Model: Species

Now fit a simpler model has an with only ExperimentNo.

##   
## Call:  
## glm(formula = log(Bd.endpoint.GE) ~ ExperimentNo, family = gaussian,   
## data = Bd.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -5.6547 -0.7265 0.2550 0.8231 3.0062   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.8691 0.1765 21.927 < 2e-16 \*\*\*  
## ExperimentNo2 -0.2020 0.2474 -0.816 0.415249   
## ExperimentNo3 -3.7096 0.3769 -9.843 < 2e-16 \*\*\*  
## ExperimentNo4 -1.7464 0.2748 -6.356 1.48e-09 \*\*\*  
## ExperimentNo5 -1.2223 0.3196 -3.825 0.000177 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 1.774683)  
##   
## Null deviance: 573.05 on 196 degrees of freedom  
## Residual deviance: 340.74 on 192 degrees of freedom  
## AIC: 679  
##   
## Number of Fisher Scoring iterations: 2



The AIC suggests you don’t gain much explanatory power by lossing a parameter (Treatment).

# Part 4: Visualise Rv endpoint infection load data

**N.B.** At the moment there are 3 samples missing EBF3N qPCR data so as a temporary fix I have averaged the EBF3N score for all the samples and used that to calculate a rough viral load for these 3 samples

For the plots below I only plot Rv infection loads over 0 and filter out one extremely high Rv viral load (greater than 2000).

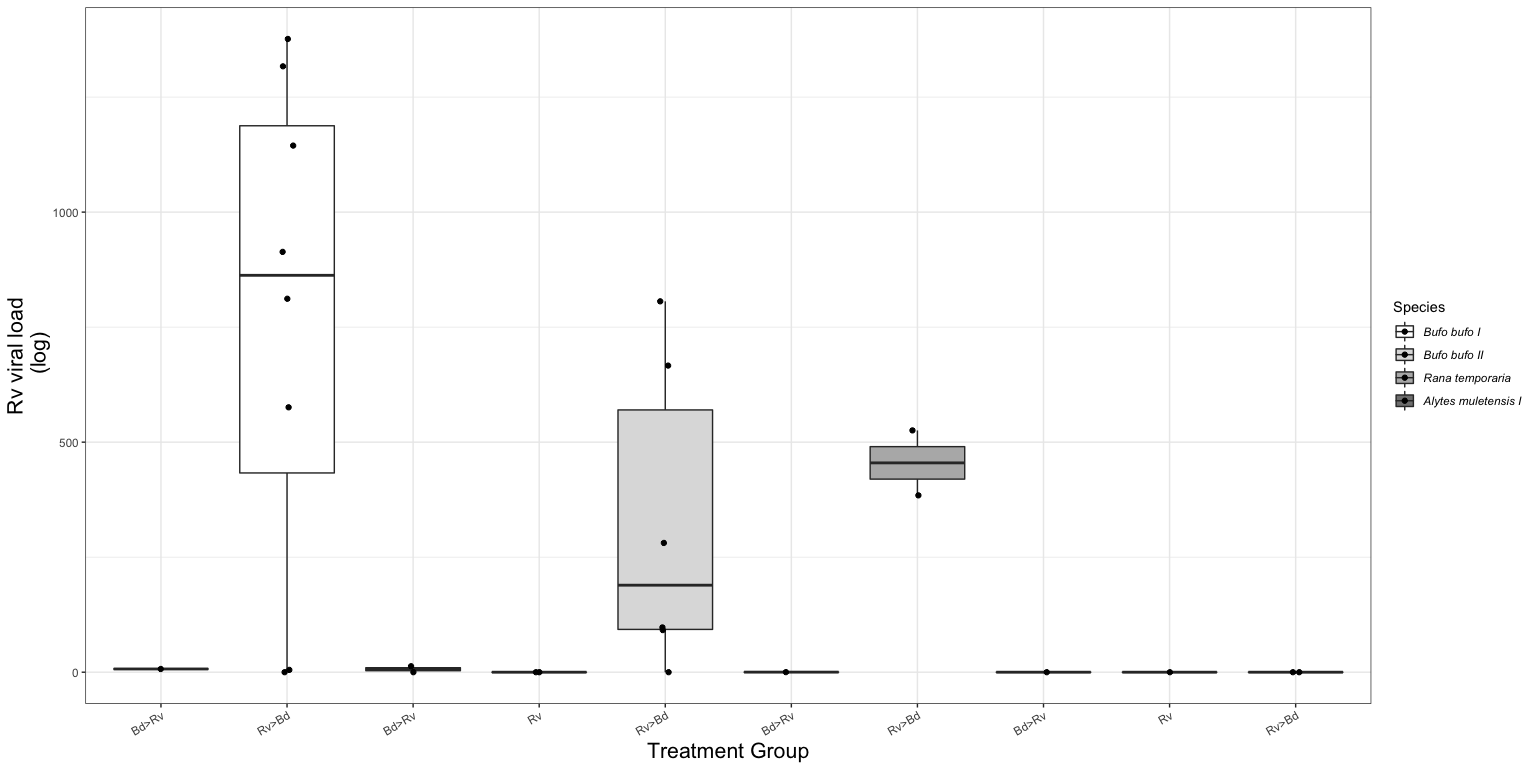


Fig.3. Boxplot of the endpoint Ranaviral load for each treatment group across the three host species. Viral load has been normalised using Leung et al.’s (2017) method. The black dots represent each sample.

?? **QUESTION** is it ok to do this? Or am I better often doing log(n+1)? My reasoning was that we are not interested in no infection!?

The distribution of logged Rv viral load.

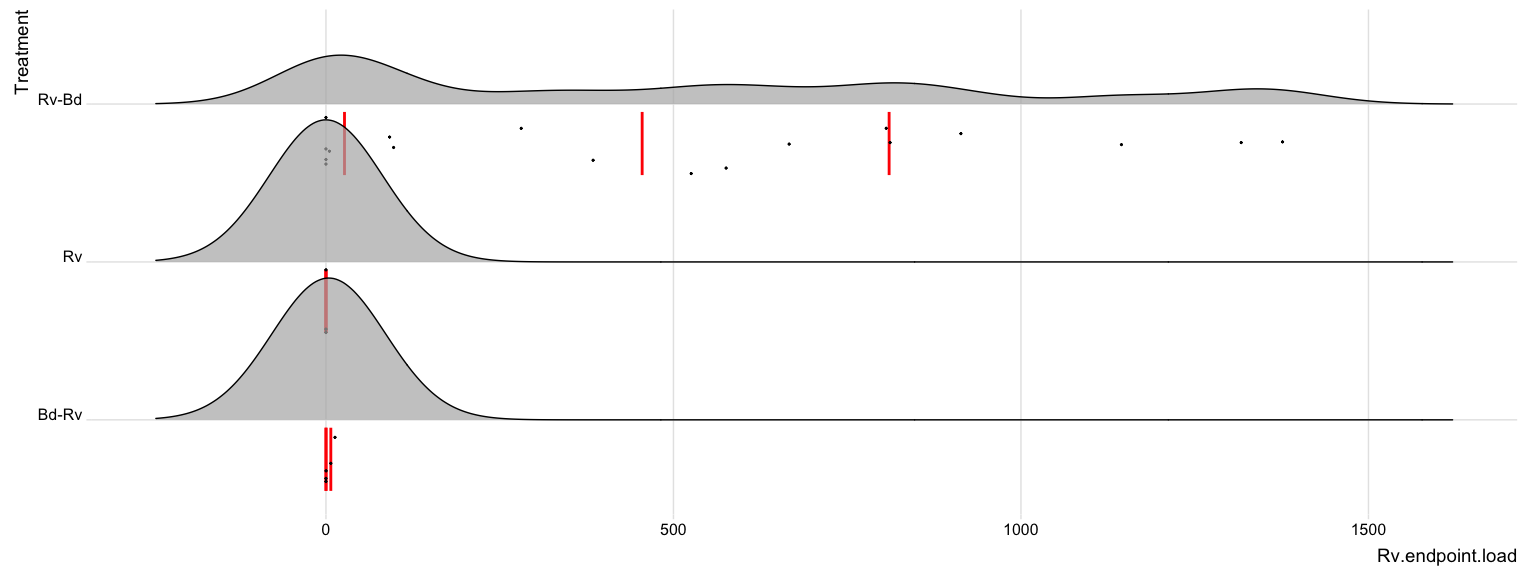


Fig. 4a. The distribution of Rv load from endpoint tissue samples.

…and when Rv viral load is logged because the distribution of Rv load is skewed.

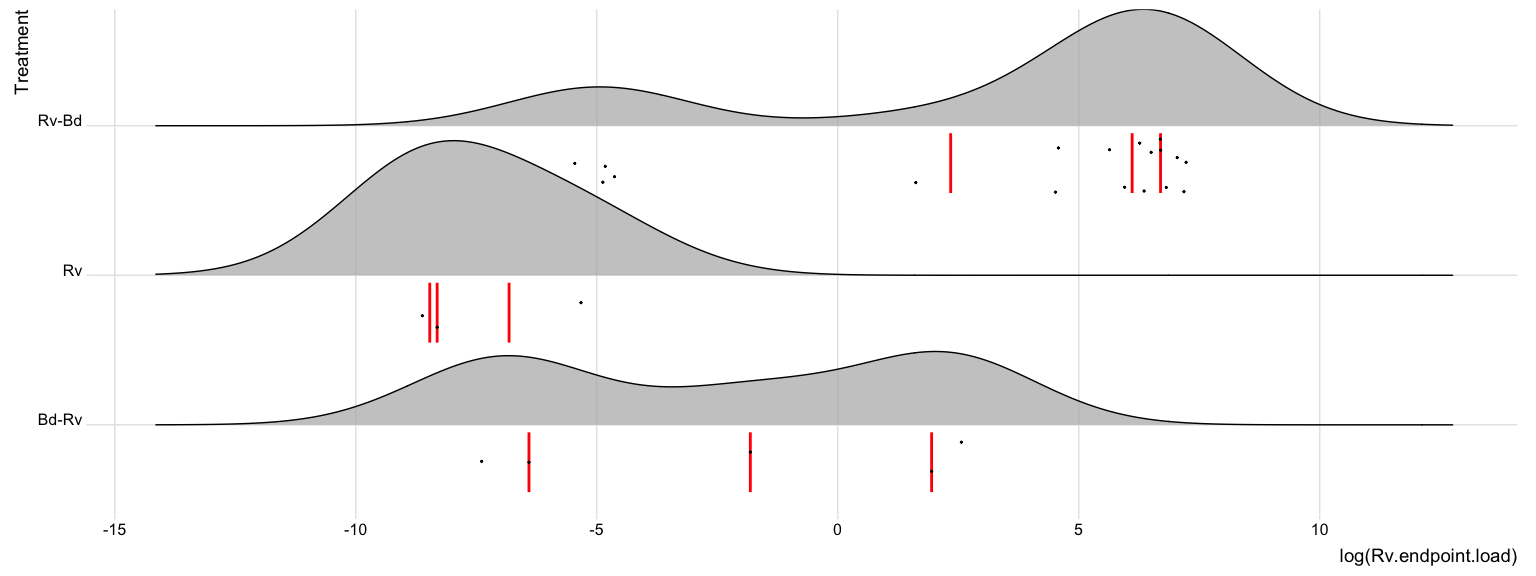


Fig. 4b. The logged distribution of Rv load from endpoint tissue samples.

## Part 5: Endpoint Infection Status: Rv

Here I create a dataframe with only individuals that am infected with Rv

?? **QUESTION** is there a threshold for ranavirus infection ???

Rv.load <- data.endpoint %>%  
 filter((Rv.endpoint.load > 0.1)) %>%   
 select(ID, Species, ExperimentNo, Scenario, Treatment, Rv.MCPendpoint.status, Rv.EBF3Nendpoint.status, Rv.endpoint.load)   
  
droplevels(Rv.load)

### Model Selection:

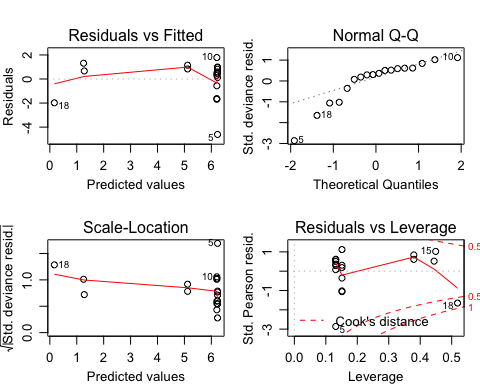
Choosing a error structure family. Again, I decided to do this with a semi-maximal model to help the fitting as much as possible.

1. gaussian distribution with data log transformed

Rv.load2.1 <- glm(log(Rv.endpoint.load) ~ Treatment + ExperimentNo, data=Rv.load, family= "gaussian")  
summary(Rv.load2.1) #AIC = 76.262 #Residual deviance: 41.834 on 14 degrees of freedom

##   
## Call:  
## glm(formula = log(Rv.endpoint.load) ~ Treatment + ExperimentNo,   
## family = "gaussian", data = Rv.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.6115 -0.3936 0.5372 0.9230 1.7846   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.27919 1.15206 1.110 0.285556   
## TreatmentRv-Bd 4.95117 1.11610 4.436 0.000564 \*\*\*  
## ExperimentNo2 -0.02557 0.89487 -0.029 0.977607   
## ExperimentNo3 -1.11255 1.19317 -0.932 0.366915   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 2.988172)  
##   
## Null deviance: 111.617 on 17 degrees of freedom  
## Residual deviance: 41.834 on 14 degrees of freedom  
## AIC: 76.262  
##   
## Number of Fisher Scoring iterations: 2

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Rv.load2.1)

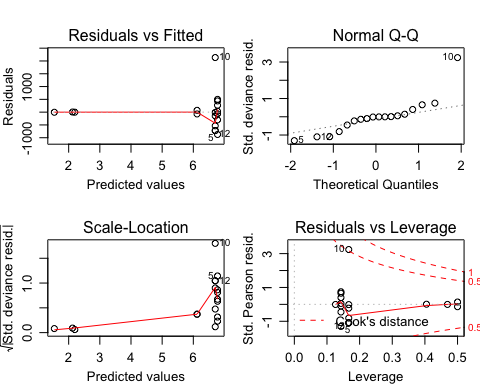
 - residuals vs fitted:  
- QQ plot is not great, (not normally distributed?) - residual vs. leverage reflects the data, low sample numbers means a few data points are influencing the model

1. gaussian distribution with log link

Rv.load2.2 <- glm(Rv.endpoint.load ~ Treatment + ExperimentNo, data=Rv.load,family="gaussian"(link='log'))  
summary(Rv.load2.2) #AIC = 293.47 #Residual deviance: 7283161 on 14 degrees of freedom degrees of freedom

##   
## Call:  
## glm(formula = Rv.endpoint.load ~ Treatment + ExperimentNo, family = gaussian(link = "log"),   
## data = Rv.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -872.65 -263.69 -6.86 61.93 2134.21   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 2.19702 54.84766 0.040 0.969  
## TreatmentRv-Bd 4.58028 54.84761 0.084 0.935  
## ExperimentNo2 -0.07371 0.47634 -0.155 0.879  
## ExperimentNo3 -0.65714 1.16326 -0.565 0.581  
##   
## (Dispersion parameter for gaussian family taken to be 520230.8)  
##   
## Null deviance: 9123672 on 17 degrees of freedom  
## Residual deviance: 7283161 on 14 degrees of freedom  
## AIC: 293.47  
##   
## Number of Fisher Scoring iterations: 6

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Rv.load2.2)



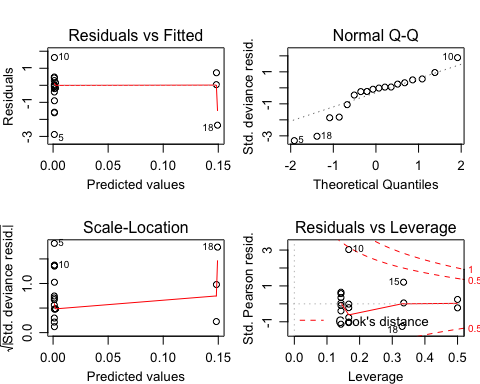
Again, this is a terrible model

1. Gamma distribution

# Gamma distribution  
Rv.load2.3 <- glm(Rv.endpoint.load ~ Treatment + ExperimentNo, data=Rv.load, family=Gamma())  
summary(Rv.load2.3) #AIC = 257.35 #Residual deviance: 23.678 on 14 degrees of freedom

##   
## Call:  
## glm(formula = Rv.endpoint.load ~ Treatment + ExperimentNo, family = Gamma(),   
## data = Rv.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.88592 -0.77833 -0.04491 0.24714 1.63001   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.483e-01 8.110e-02 1.828 0.0889 .  
## TreatmentRv-Bd -1.471e-01 8.110e-02 -1.814 0.0912 .  
## ExperimentNo2 8.565e-05 6.236e-04 0.137 0.8927   
## ExperimentNo3 1.075e-03 1.534e-03 0.700 0.4951   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for Gamma family taken to be 0.893)  
##   
## Null deviance: 46.427 on 17 degrees of freedom  
## Residual deviance: 23.678 on 14 degrees of freedom  
## AIC: 257.35  
##   
## Number of Fisher Scoring iterations: 7

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Rv.load2.3)

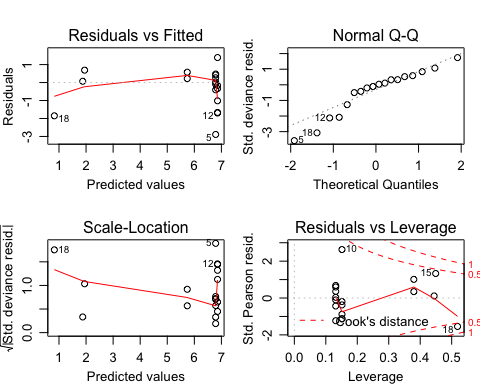
 - residuals vs fitted: patterning  
- QQ plot is not great, (not normally distributed?) - residual vs. leverage is better but still reflects the data, low sample numbers means a few data points are influencing the model

1. Gamma distribution with log-link

Rv.load2.4 <- glm(Rv.endpoint.load ~ Treatment + ExperimentNo, data=Rv.load, family="Gamma"(link='log'))  
summary(Rv.load2.4) #AIC = 255.8 #Residual deviance: 22.027 on 14 degrees of freedom

##   
## Call:  
## glm(formula = Rv.endpoint.load ~ Treatment + ExperimentNo, family = Gamma(link = "log"),   
## data = Rv.load)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.88952 -0.86174 -0.02881 0.38392 1.39261   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.87732 0.57792 3.248 0.00583 \*\*   
## TreatmentRv-Bd 4.91046 0.55988 8.771 4.62e-07 \*\*\*  
## ExperimentNo2 0.06971 0.44890 0.155 0.87881   
## ExperimentNo3 -1.04891 0.59854 -1.752 0.10156   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for Gamma family taken to be 0.7519493)  
##   
## Null deviance: 46.427 on 17 degrees of freedom  
## Residual deviance: 22.027 on 14 degrees of freedom  
## AIC: 255.8  
##   
## Number of Fisher Scoring iterations: 9

par(mfrow=c(2,2), mar=c(3,3,3,1), mgp=c(2,0.8,0))  
plot(Rv.load2.4)



?? **QUESTION** log(Rv load) and gaussian or Gamma? None of them look ideal

### Model Comparison

The four models (all using log(Bd load) and gaussian error structure)

Rv.load1 <- glm(log(Rv.endpoint.load) ~ Treatment \* ExperimentNo, data=Rv.load, family=gaussian)  
  
Rv.load2 <- glm(log(Rv.endpoint.load) ~ Treatment + ExperimentNo, data=Rv.load, family=gaussian)  
  
Rv.load3 <- glm(log(Rv.endpoint.load) ~ Treatment, data=Rv.load, family=gaussian)  
  
Rv.load4 <- glm(log(Rv.endpoint.load) ~ ExperimentNo, data=Rv.load, family=gaussian)

?? **QUESTION** should I change the test = to better reflect the data type?

anova(Rv.load2, Rv.load4, test="Chisq") # compares Trt and Sp to just Sp

## Analysis of Deviance Table  
##   
## Model 1: log(Rv.endpoint.load) ~ Treatment + ExperimentNo  
## Model 2: log(Rv.endpoint.load) ~ ExperimentNo  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)   
## 1 14 41.834   
## 2 15 100.639 -1 -58.805 9.16e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

anova(Rv.load2, Rv.load3, test="Chisq") # compares Trt and Sp to just Trt

## Analysis of Deviance Table  
##   
## Model 1: log(Rv.endpoint.load) ~ Treatment + ExperimentNo  
## Model 2: log(Rv.endpoint.load) ~ Treatment  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)  
## 1 14 41.834   
## 2 16 44.743 -2 -2.9089 0.6146

anova(Rv.load1, Rv.load2, test="Chisq") # compare more complex model to one with interaction

## Analysis of Deviance Table  
##   
## Model 1: log(Rv.endpoint.load) ~ Treatment \* ExperimentNo  
## Model 2: log(Rv.endpoint.load) ~ Treatment + ExperimentNo  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)  
## 1 12 33.432   
## 2 14 41.834 -2 -8.402 0.2214

ANOVA 1: suggests we should favour the more complex model (Treatment + ExperimentNo) over the model with ExperimentNo only (pvalue = < .001)

ANOVA 2: suggests we should reject the more complex model (Treatment + ExperimentNo) in favour of the model with Treatment only (pvalue = 0.6146) as adding ExperimentNo didn’t lead to significantly improved fit

ANOVA 3: shows adding a interaction term between Treatment and ExperimentNo did not significantly imporve fit (p-value = 0.2214)

**Conclusion**: we should choose the model with just Treatment (like Rv status)

### Model Fit:

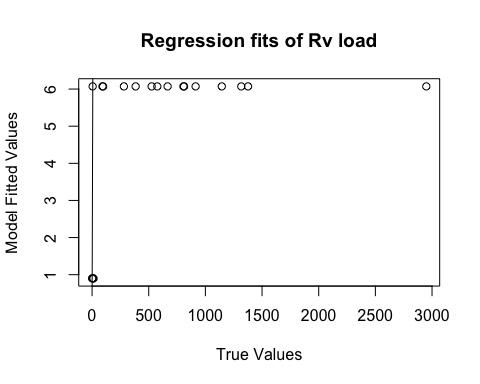


Fig. 6. Plot of relationship between the fitted values from the model and the true data values.

Errmmmmm…… ??

### Model Plotting:

Rv.load3 <- glm(log(Rv.endpoint.load) ~ Treatment, data=Rv.load, family=gaussian)  
  
# create a dataframe of "new" data   
newdat <- expand.grid(ExperimentNo=c("1", "2", "3", "4", "5"),Treatment=c( "Bd-Rv", "Rv-Bd"))  
  
# predict the value/result of the new data using the glm  
Rv.newdat <-cbind(newdat, predict(object = Rv.load3, # the model   
 newdata=newdat, se=TRUE, type="response", print.matrix=T)) # dataframe of new data   
Rv.newdat

## ExperimentNo Treatment fit se.fit residual.scale  
## 1 1 Bd-Rv 0.8998199 0.9654808 1.672262  
## 2 2 Bd-Rv 0.8998199 0.9654808 1.672262  
## 3 3 Bd-Rv 0.8998199 0.9654808 1.672262  
## 4 4 Bd-Rv 0.8998199 0.9654808 1.672262  
## 5 5 Bd-Rv 0.8998199 0.9654808 1.672262  
## 6 1 Rv-Bd 6.0717971 0.4317761 1.672262  
## 7 2 Rv-Bd 6.0717971 0.4317761 1.672262  
## 8 3 Rv-Bd 6.0717971 0.4317761 1.672262  
## 9 4 Rv-Bd 6.0717971 0.4317761 1.672262  
## 10 5 Rv-Bd 6.0717971 0.4317761 1.672262

expl.var <- c(1:2) # chose the range for the x-axis (Treatment)  
exp.labs <- c("1" = "Bufo bufo I", "2" = "Bufo bufo II", "3" = "Rana temporaria", "4" = "Alytes muletensis I")  
  
newdat1<- subset(Rv.newdat, ExperimentNo== "1") # need to subset the data so you can plot each seperatly   
newdat2<- subset(Rv.newdat, ExperimentNo=="2")  
newdat3<- subset(Rv.newdat, ExperimentNo=="3")  
newdat4<- subset(Rv.newdat, ExperimentNo=="4")

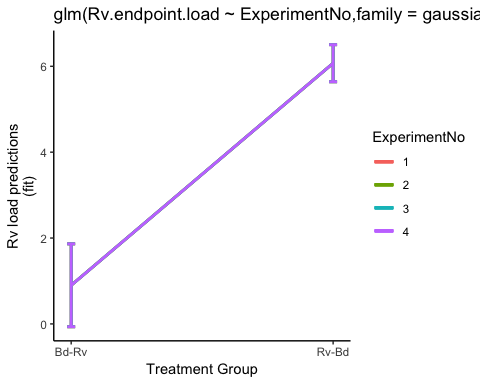


Fig. 7. Probability of Rv infection load predicted by model