Size-Driven Preservational and Macroecological Biases in the Latest Maastrichtian terrestrial vertebrate assemblages of North America

Caleb M. Brown, Nicolás E. Campione, Gregory P. Wilson Mantilla, and David C. Evans

code compiled by: Caleb M. Brown ([caleb.brown@gov.ab.ca](mailto:caleb.brown@gov.ab.ca)) and Nicolás E. Campione ([ncampion@une.edu.au](mailto:ncampion@une.edu.au))

The following code accompanies the aforementioned manuscript and includes all analyses. N.B. The code should run as is, assuming all files are in the same working directory, however file names may need to be adjusted.

# Set-up

## Required libraries

#install.packages(c("e1071", "oce")) #run if installation is required  
library(e1071)  
library(oce)

# Hell Creek Dinosaurs

## Data sets

The primary and skew data objects depend on the active HC or skewer data set, respectively, that are read into R. The data sets reflect minor differences in construction, as explained in the manuscript. The # symbol should be toggled to access the other forms of the data.

#Primary Hell Creek and Lance data  
HC <- HC\_split\_data <- read.csv("HC\_split\_data.csv")  
HC <- HC\_lump\_data <- read.csv("HC\_lump\_data.csv")  
HC <- HCL\_split\_data <- read.csv("HCL\_split\_data.csv")  
HC <- HCL\_lump\_data <- read.csv("HCL\_lump\_data.csv")  
orn <- HC[HC$Clade == "Ornithischia",] # Ornithischians  
ther <- HC[HC$Clade == "Theropoda",] # Theropods  
dim(HC)

## [1] 33 7

#Rate data  
rate <- Rate\_data <- read.csv("Rate\_data.csv")  
  
#Hell Creek survey data  
hell <- Survey\_data\_updated <- read.csv("Survey\_data\_updated.csv")  
  
#Skew data  
skewer <- HCL\_split\_skew\_data <- read.csv("HCL\_split\_skew\_data.csv")  
skewer <- HC\_lump\_skew <- read.csv("HC\_lump\_skew.csv")  
skewer <- DPP\_skew\_data <- read.csv("DPP\_skew\_data.csv")  
skewer2 <- Skew\_date <- read.csv("skew\_date.csv")

## Mass vs. Completeness

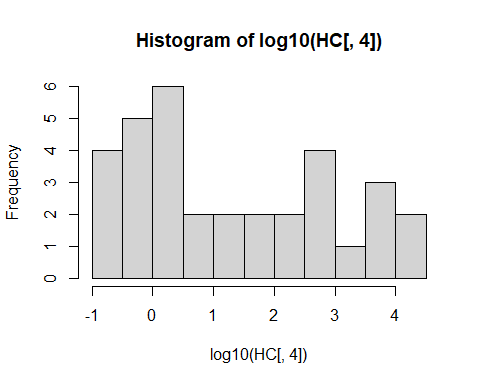
#Normality tests  
shapiro.test(log10(HC[,4])) #body mass

##   
## Shapiro-Wilk normality test  
##   
## data: log10(HC[, 4])  
## W = 0.90436, p-value = 0.006926

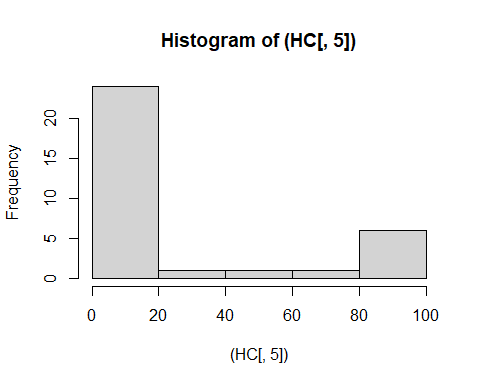
shapiro.test((HC[,5])) #completeness

##   
## Shapiro-Wilk normality test  
##   
## data: (HC[, 5])  
## W = 0.6127, p-value = 3.934e-08

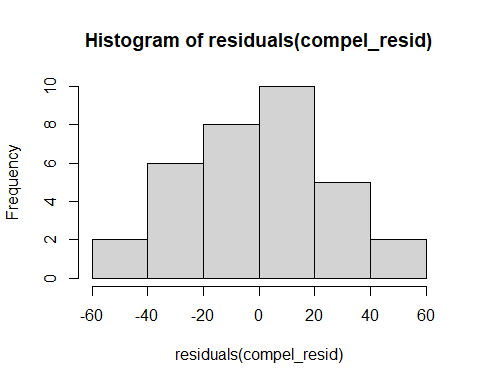
#Distributions  
hist(log10(HC[,4]))



hist((HC[,5]))



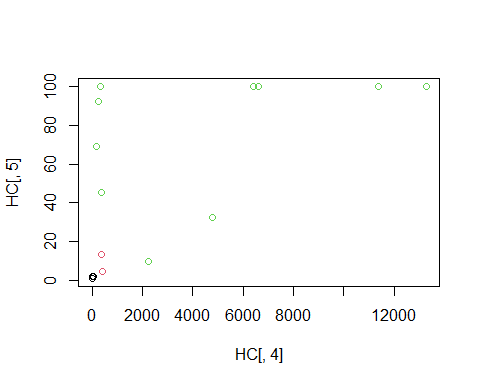
#linear models  
##mass vs. completeness  
compel\_resid <- lm(HC[,5]~log10(HC[,4]))  
hist(residuals(compel\_resid))



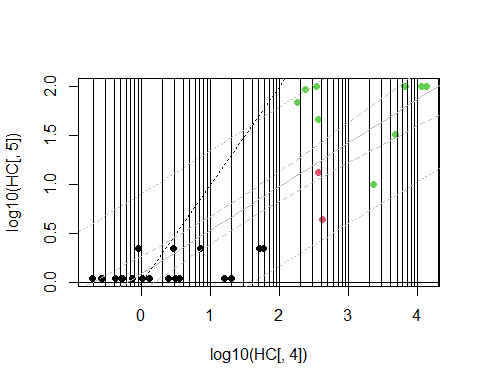
shapiro.test(residuals(compel\_resid))

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(compel\_resid)  
## W = 0.98424, p-value = 0.9014

#Bivariate Plot: mass vs. completeness  
plot(HC[,4],HC[,5],col=HC[,6])



plot(log10(HC[,4]),log10(HC[,5]),col=HC[,6],pch=19)  
##log lines  
abline(v=log10(seq(0,1,.1)))  
abline(v=log10(seq(1,9,1)))  
abline(v=log10(seq(10,90,10)))  
abline(v=log10(seq(100,900,100)))  
abline(v=log10(seq(1000,9000,1000)))  
abline(v=log10(seq(10000,90000,10000)))  
abline(h=log10(seq(0.1,1,.1)))  
abline(h=log10(seq(.01,.09,.01)))  
  
#Bivariate linear model: mass vs. completeness  
x <-log10(HC[,4])  
y <-log10(HC[,5])  
xy <- data.frame(x, y)  
mdl <- lm(y ~ x, data = xy)  
predx <- data.frame(x = seq(from = -1, to = 5, by = 0.1))  
conf.int <- cbind(predx, predict(mdl, newdata = predx, interval = "confidence", level = 0.95))  
pred.int <- cbind(predx, predict(mdl, newdata = predx, interval = "prediction", level = 0.95))  
points(conf.int[,1],conf.int[,2],typ="l",col="grey")  
points(conf.int[,1],conf.int[,3],typ="l",lty=2,col="grey")  
points(conf.int[,1],conf.int[,4],typ="l",lty=2,col="grey")  
points(pred.int[,1],pred.int[,3],typ="l",lty=3,col="grey")  
points(pred.int[,1],pred.int[,4],typ="l",lty=3,col="grey")  
abline(0,1,lty=3)



summary(mdl)

##   
## Call:  
## lm(formula = y ~ x, data = xy)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.62437 -0.21911 0.04938 0.21069 0.81568   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.09147 0.08846 1.034 0.309   
## x 0.44444 0.04340 10.241 1.81e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3898 on 31 degrees of freedom  
## Multiple R-squared: 0.7719, Adjusted R-squared: 0.7645   
## F-statistic: 104.9 on 1 and 31 DF, p-value: 1.808e-11

confint(mdl)

## 2.5 % 97.5 %  
## (Intercept) -0.08894955 0.2718875  
## x 0.35592931 0.5329449

#Correlation tests  
##all dinosaurs  
cor.test(log10(HC[,4]),log10(HC[,5]))

##   
## Pearson's product-moment correlation  
##   
## data: log10(HC[, 4]) and log10(HC[, 5])  
## t = 10.241, df = 31, p-value = 1.808e-11  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.7664124 0.9387303  
## sample estimates:  
## cor   
## 0.8785584

cor.test(log10(HC[,4]),HC[,5],method="pearson")

##   
## Pearson's product-moment correlation  
##   
## data: log10(HC[, 4]) and HC[, 5]  
## t = 6.5688, df = 31, p-value = 2.459e-07  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.5683827 0.8765804  
## sample estimates:  
## cor   
## 0.7628383

cor.test(log10(HC[,4]),HC[,5],method="kendall")

## Warning in cor.test.default(log10(HC[, 4]), HC[, 5], method = "kendall"): Cannot  
## compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: log10(HC[, 4]) and HC[, 5]  
## z = 5.306, p-value = 1.121e-07  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.7083391

range(HC[,4])

## [1] 0.192 13274.610

dim(HC)

## [1] 33 7

##ornithischians  
cor.test(log10(orn[,4]),orn[,5],method="pearson")

##   
## Pearson's product-moment correlation  
##   
## data: log10(orn[, 4]) and orn[, 5]  
## t = 1.9283, df = 7, p-value = 0.09516  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.1233857 0.9007699  
## sample estimates:  
## cor   
## 0.5890005

cor.test(log10(orn[,4]),orn[,5],method="kendall")

## Warning in cor.test.default(log10(orn[, 4]), orn[, 5], method = "kendall"):  
## Cannot compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: log10(orn[, 4]) and orn[, 5]  
## z = 1.7527, p-value = 0.07965  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.4868645

range(orn[,4])

## [1] 16.14 13274.61

dim(orn)

## [1] 9 7

##saurischians  
cor.test(log10(ther[,4]),ther[,5],method="pearson")

##   
## Pearson's product-moment correlation  
##   
## data: log10(ther[, 4]) and ther[, 5]  
## t = 5.3264, df = 22, p-value = 2.399e-05  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.4977997 0.8857347  
## sample estimates:  
## cor   
## 0.7504899

cor.test(log10(ther[,4]),ther[,5],method="kendall")

## Warning in cor.test.default(log10(ther[, 4]), ther[, 5], method = "kendall"):  
## Cannot compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: log10(ther[, 4]) and ther[, 5]  
## z = 3.8985, p-value = 9.68e-05  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.6427551

range(ther[,4])

## [1] 0.192 6403.250

dim(ther)

## [1] 24 7

# Mean/Median completeness for the size classes...  
mode(HC[c(1:25),5])

## [1] "numeric"

large\_comp <- HC[c(22:33),5]  
small\_comp <- HC[c(1:21),5]  
mean(small\_comp)

## [1] 1.374286

median(small\_comp)

## [1] 1.11

mean(large\_comp)

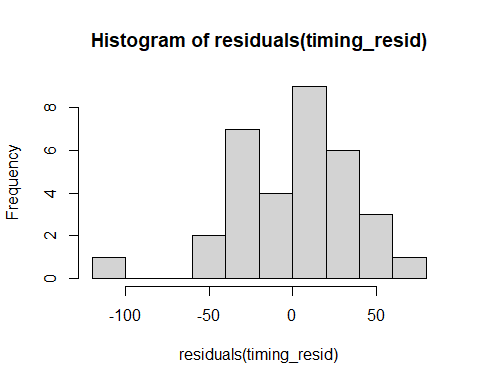
## [1] 63.88917

t.test(large\_comp,small\_comp)

##   
## Welch Two Sample t-test  
##   
## data: large\_comp and small\_comp  
## t = 5.4011, df = 11.002, p-value = 0.0002162  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## 37.04002 87.98974  
## sample estimates:  
## mean of x mean of y   
## 63.889167 1.374286

## Mass vs. Timing

timing\_resid<-lm(HC[,3]~log10(HC[,4]))  
hist(residuals(timing\_resid))



shapiro.test(residuals(timing\_resid))

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(timing\_resid)  
## W = 0.9563, p-value = 0.2027

date <- HC[c(26:43),2] # adjusted membership based on size classes  
time <- date-1882  
large\_time <- time  
small\_time <- time  
median(small\_time, na.rm = TRUE)

## [1] 24.5

t.test(large\_time,small\_time)

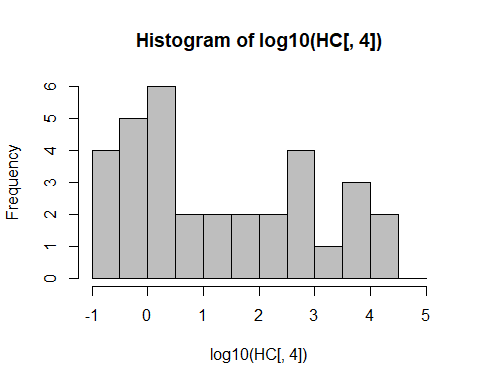
##   
## Welch Two Sample t-test  
##   
## data: large\_time and small\_time  
## t = 0, df = 14, p-value = 1  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -43.9735 43.9735  
## sample estimates:  
## mean of x mean of y   
## 37.375 37.375

length(small\_time)

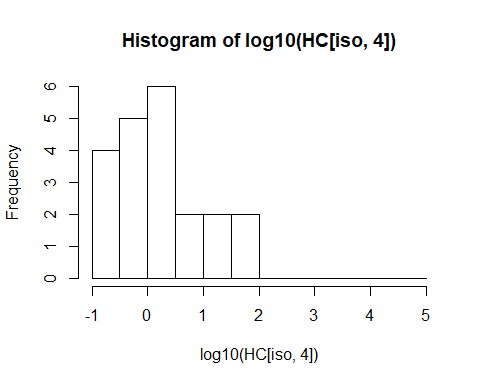
## [1] 18

## Mass vs. Taphonomic Mode

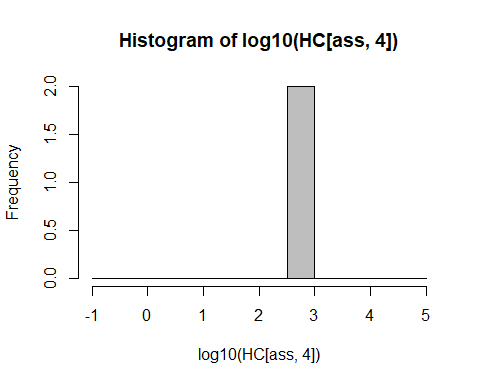
#taphonomic modes  
iso <- HC[,6]==1  
ass <- HC[,6]==2  
art <- HC[,6]==3  
##general distribution  
hist(log10(HC[,4]),breaks=c(seq(-1,5,.5)),col="grey")



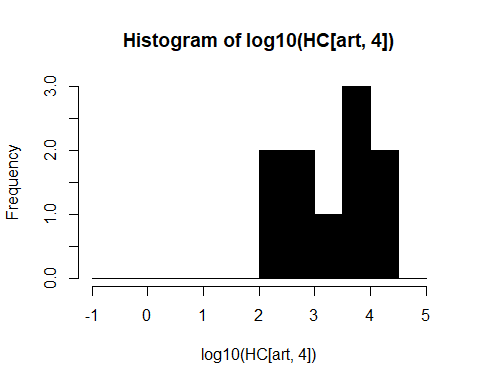
##mode distributions  
hist(log10(HC[iso,4]),breaks=c(seq(-1,5,.5)),col="white")



hist(log10(HC[ass,4]),breaks=c(seq(-1,5,.5)),col="grey")



hist(log10(HC[art,4]),breaks=c(seq(-1,5,.5)),col="black")



#Spearman correlations  
cor.test(HC[,6],HC[,4],method="kendall")

## Warning in cor.test.default(HC[, 6], HC[, 4], method = "kendall"): Cannot  
## compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: HC[, 6] and HC[, 4]  
## z = 4.7569, p-value = 1.966e-06  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.6820911

cor.test(HC[,6],HC[,4],method="spearman")

## Warning in cor.test.default(HC[, 6], HC[, 4], method = "spearman"): Cannot  
## compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: HC[, 6] and HC[, 4]  
## S = 1022.9, p-value = 2.524e-09  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.8290578

#Chi.square test between modes, given mass  
big <- c(0,3,15)  
small <- c(23,1,1)  
both <- rbind(big,small)  
chisq.test(big,small)

## Warning in chisq.test(big, small): Chi-squared approximation may be incorrect

##   
## Pearson's Chi-squared test  
##   
## data: big and small  
## X-squared = 3, df = 2, p-value = 0.2231

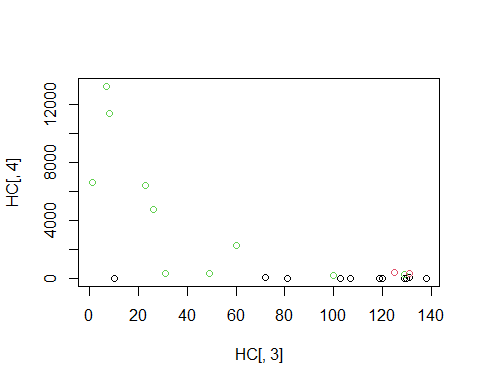
chisq.test(both)

## Warning in chisq.test(both): Chi-squared approximation may be incorrect

##   
## Pearson's Chi-squared test  
##   
## data: both  
## X-squared = 36.066, df = 2, p-value = 1.473e-08

## Mass vs. Timing of Description

#bivariate plot  
plot(HC[,3],HC[,4],col=HC[,6])



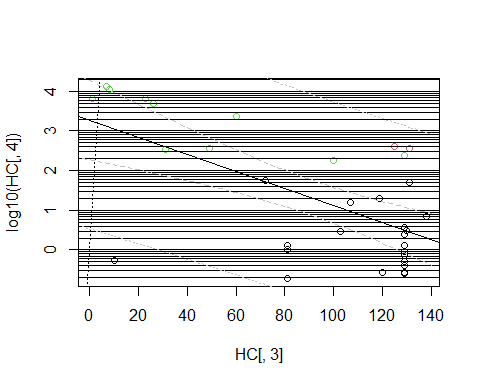
plot(HC[,3],log10(HC[,4]),col=HC[,6])  
abline(h=log10(70))  
#log lines  
abline(h=log10(seq(0,1,.1)))  
abline(h=log10(seq(1,9,1)))  
abline(h=log10(seq(10,90,10)))  
abline(h=log10(seq(100,900,100)))  
abline(h=log10(seq(1000,9000,1000)))  
abline(h=log10(seq(10000,90000,10000)))  
#curves  
x <-HC[,3]  
y <-log10(HC[,4])  
cor.test(x,y)

##   
## Pearson's product-moment correlation  
##   
## data: x and y  
## t = -4.4928, df = 31, p-value = 9.137e-05  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.7990241 -0.3629070  
## sample estimates:  
## cor   
## -0.6279768

predict(lm(y ~ x))

## 1 2 3 4 5 6 7 8   
## 1.5360455 0.5057107 0.6988985 0.5057107 0.5057107 0.5057107 3.0600823 0.5057107   
## 9 10 11 12 13 14 15 16   
## 0.5057107 1.5360455 1.5360455 0.5057107 0.5057107 1.0638087 0.4842454 0.5057107   
## 17 18 19 20 21 22 23 24   
## 0.3125229 0.9779475 0.7203638 0.4627801 1.7292332 1.1282046 0.5057107 2.6093109   
## 25 26 27 28 29 30 31 32   
## 0.4627801 2.2229353 0.5915719 1.9868169 2.7166374 2.7810333 3.2532701 3.1030130   
## 33   
## 3.1244783

xy <- data.frame(x, y)  
mdl <- lm(y ~ x, data = xy)  
predx <- data.frame(x = seq(from = -3, to = 140, by = 1))  
conf.int <- cbind(predx, predict(mdl, newdata = predx, interval = "confidence", level = 0.95))  
pred.int <- cbind(predx, predict(mdl, newdata = predx, interval = "prediction", level = 0.95))  
abline(mdl)  
points(conf.int[,1],conf.int[,3],typ="l",lty=2,col="grey")  
points(conf.int[,1],conf.int[,4],typ="l",lty=2,col="grey")  
points(pred.int[,1],pred.int[,3],typ="l",lty=3,col="grey")  
points(pred.int[,1],pred.int[,4],typ="l",lty=3,col="grey")  
abline(0,1,lty=3)



summary(mdl)

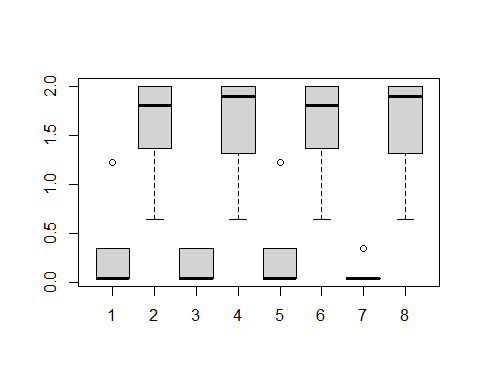
##   
## Call:  
## lm(formula = y ~ x, data = xy)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.3342 -0.7981 0.0302 0.9629 2.0934   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.274735 0.489337 6.692 1.74e-07 \*\*\*  
## x -0.021465 0.004778 -4.493 9.14e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.256 on 31 degrees of freedom  
## Multiple R-squared: 0.3944, Adjusted R-squared: 0.3748   
## F-statistic: 20.19 on 1 and 31 DF, p-value: 9.137e-05

confint(mdl)

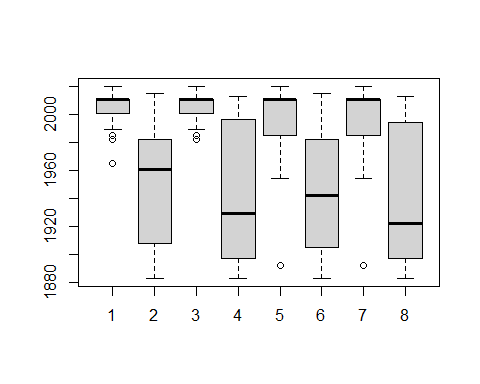
## 2.5 % 97.5 %  
## (Intercept) 2.27672562 4.27274521  
## x -0.03120956 -0.01172106

## Boxplots

#relative to body mass  
boxplot(log10(HC\_split\_data[c(1:17),5]),log10(HC\_split\_data[c(18:35),5]),log10(HC\_lump\_data[c(1:13),5]\*100),log10(HC\_lump\_data[c(14:25),5]\*100),  
 log10(HCL\_split\_data[c(1:25),5]),log10(HCL\_split\_data[c(26:43),5]),log10(HCL\_lump\_data[c(1:21),5]),  
 log10(HCL\_lump\_data[c(22:33),5]))

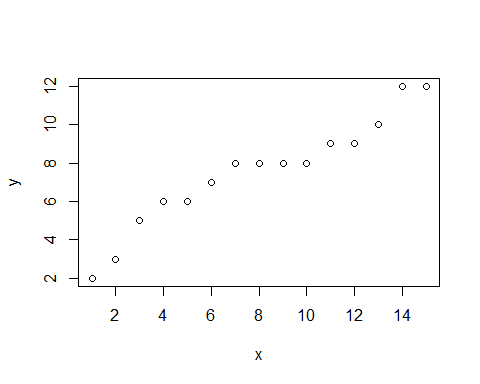


#relative to year  
boxplot(HC\_split\_data[c(1:17),2],HC\_split\_data[c(18:35),2],HC\_lump\_data[c(1:13),2],HC\_lump\_data[c(14:25),2],  
 HCL\_split\_data[c(1:25),2],HCL\_split\_data[c(26:43),2],HCL\_lump\_data[c(1:21),2],HCL\_lump\_data[c(22:33),2])



## Rate Analysis

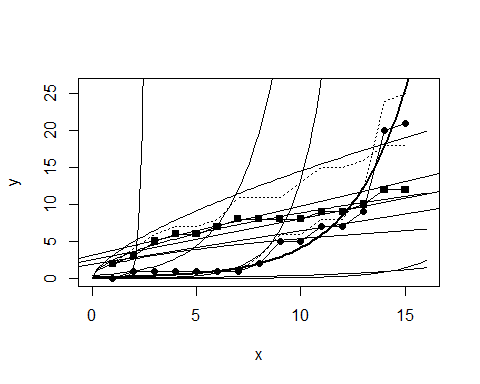
x <- rate[,1]  
y <- rate[,10]  
y2 <- rate[,9]  
y3 <- rate[,8]  
y4 <- rate[,7]  
  
plot(x,y)



plot(x,y,xlim=c(0,16),ylim=c(0,26), pch=15)  
points(x,y2,pch=19)  
points(x,y,typ="l")  
points(x,y2,typ="l")  
points(x,y3,typ="l",lty=3)  
points(x,y4,typ="l",lty=3)  
  
abline(2.4684,0.5555)  
abline(1.9025864,0.4513216)  
abline(3.1114595,0.6697293)  
line1 <- lm(y~x)  
confint(line1)

## 2.5 % 97.5 %  
## (Intercept) 1.6938262 3.5442690  
## x 0.5125249 0.7160465

curve((2.4684\*(x^0.5555)),add=T)  
curve((1.9025864\*(x^0.4513216)),add=T)  
curve((3.1114595\*(x^0.6697293)),add=T)  
  
curve((0.1075\*exp(x\*0.365)),add=T,lwd=2)  
curve((0.05120684\*exp(x\*0.20748350)),add=T)  
curve((0.3630822\*exp(x\*0.4997171)),add=T)  
  
curve((0.00000529\*exp(x\*6.314)),add=T)  
curve((0.0005802719\*exp(x\*0.5200346758)),add=T)  
curve((0.009654995\*exp(x\*0.723677821)),add=T)



## Model Fitting

plot(x,y)  
  
lin <- nls(y~(m\*x+b),start=list(m=0.004,b=3.1))  
log<-nls(y~(m\*log(x)+b),start=list(m=1.34,b=2.75))  
pow<-nls(y~(m\*(x^b)),start=list(m=2.1,b=0.78))  
expo<-nls(y~(m\*exp(x\*b)),start=list(m=0.2,b=0.21))  
summary(lin)

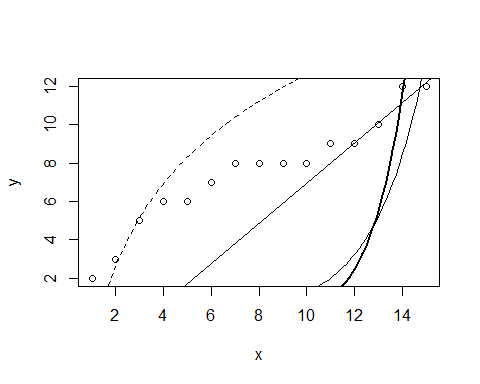
##   
## Formula: y ~ (m \* x + b)  
##   
## Parameters:  
## Estimate Std. Error t value Pr(>|t|)   
## m 0.6143 0.0471 13.041 7.66e-09 \*\*\*  
## b 2.6191 0.4283 6.115 3.69e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7882 on 13 degrees of freedom  
##   
## Number of iterations to convergence: 1   
## Achieved convergence tolerance: 5.927e-07

confint(expo)

## Waiting for profiling to be done...

## 2.5% 97.5%  
## m 3.0940175 4.61924661  
## b 0.0610637 0.09628681

abline(-3.505,1.048)  
  
curve(6.178\*log(x)+-1.623, add=TRUE,lty=2)  
curve((0.01088\*(x^0.995)),add=T)  
curve((0.01088\*exp(x\*0.4750)),add=T)  
  
curve((4.589e-11\*(x^9.947)),add=T,lwd=2)



AIC(lin)

## [1] 39.28121

AIC(log)

## [1] 41.66868

AIC(pow)

## [1] 34.76541

AIC(expo)

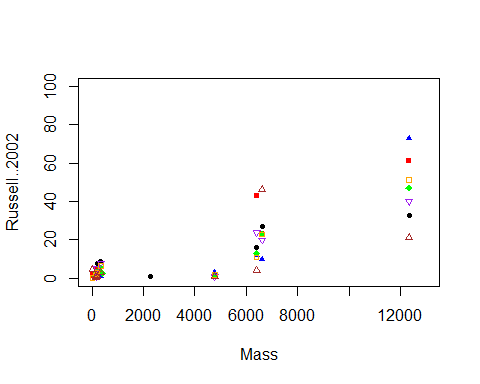
## [1] 45.81723

## Hell Creek Survey Data

attach(hell)  
head(hell)

## X Mass X.1 Russell..2002 White.et.al...1998  
## 1 Ceratopsidae 12334.16 4.091109 33 61  
## 2 Hadrosauridae 6610.08 3.820207 27 23  
## 3 Tyrannosauridae 6403.25 3.806400 16 43  
## 4 Thescelosauridae 338.51 2.529572 9 3  
## 5 Pachycephalosauridae 193.02 2.285602 8 1  
## 6 Ornithomimidae 178.51 2.251663 3 5  
## Lehman..1987 Horner.and.Goodwin Lyson.and.Longrich..2011  
## 1 73 40 46.9  
## 2 10 20 23.3  
## 3 12 24 13.1  
## 4 1 8 5.5  
## 5 1 5 3.2  
## 6 NA 1 2.9  
## Stein..2019.artic.assoc Stein..2019.all X.2  
## 1 51.3 21.4199432   
## 2 22.8 46.1285257   
## 3 10.9 4.2127197   
## 4 6.4 3.0802682   
## 5 2.8 0.8999215   
## 6 2.0 1.5371142

#plot  
plot(Mass,Russell..2002,col="black",pch=19, cex=0.7,xlim=c(1,13000),ylim=c(0,100))  
points(Mass,White.et.al...1998,col="red",pch=15, cex=0.7)  
points(Mass,Lehman..1987,col="blue",pch=17, cex=0.7)  
points(Mass,Lyson.and.Longrich..2011,col="green",pch=18, cex=1)  
points(Mass,Horner.and.Goodwin,col="purple",pch=25, cex=0.7)  
points(Mass,Stein..2019.artic.assoc,col="orange",pch=22, cex=0.7)  
points(Mass,Stein..2019.all,col="brown",pch=24, cex=0.7)



#plot (log transformed)  
plot(X.1,Russell..2002,col="black",pch=19, cex=0.7,xlim=c(1,5),ylim=c(0,100))  
points(X.1,White.et.al...1998,col="red",pch=15, cex=0.7)  
points(X.1,Lehman..1987,col="blue",pch=17, cex=0.7)  
points(X.1,Lyson.and.Longrich..2011,col="green",pch=18, cex=1)  
points(X.1,Horner.and.Goodwin,col="purple",pch=25, cex=0.7)  
points(X.1,Stein..2019.artic.assoc,col="orange",pch=22, cex=0.7)  
points(X.1,Stein..2019.all,col="brown",pch=24, cex=0.7)  
##correlation tests  
###Using Pearsons's  
cor.test(X.1,Russell..2002)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and Russell..2002  
## t = 2.1605, df = 7, p-value = 0.06756  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.05452574 0.91307721  
## sample estimates:  
## cor   
## 0.6324999

abline(lm(Russell..2002~X.1),col="black")  
cor.test(X.1,White.et.al...1998)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and White.et.al...1998  
## t = 4.2794, df = 6, p-value = 0.005209  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.4201761 0.9757887  
## sample estimates:  
## cor   
## 0.867881

abline(lm(White.et.al...1998~X.1),col="red")  
cor.test(X.1,Lehman..1987)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and Lehman..1987  
## t = 1.4166, df = 4, p-value = 0.2296  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.4399352 0.9458682  
## sample estimates:  
## cor   
## 0.5779875

abline(lm(Lehman..1987~X.1),col="blue")  
cor.test(X.1,Horner.and.Goodwin)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and Horner.and.Goodwin  
## t = 2.3476, df = 5, p-value = 0.06575  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.06368377 0.95591043  
## sample estimates:  
## cor   
## 0.7241004

abline(lm(Horner.and.Goodwin~X.1),col="green")  
cor.test(X.1,Lyson.and.Longrich..2011)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and Lyson.and.Longrich..2011  
## t = 2.7537, df = 7, p-value = 0.02835  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.1093284 0.9366563  
## sample estimates:  
## cor   
## 0.7210934

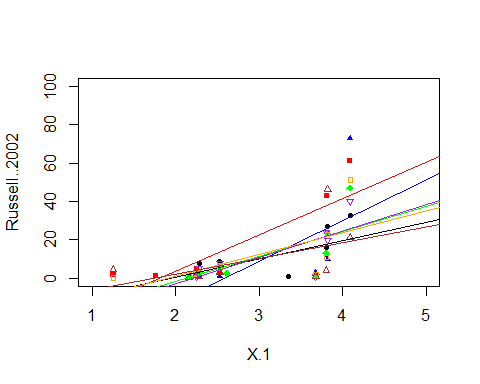
abline(lm(Lyson.and.Longrich..2011~X.1),col="purple")  
cor.test(X.1,Stein..2019.artic.assoc)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and Stein..2019.artic.assoc  
## t = 2.3986, df = 7, p-value = 0.04757  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.01359422 0.92373413  
## sample estimates:  
## cor   
## 0.6716521

abline(lm(Stein..2019.artic.assoc~X.1),col="orange")  
cor.test(X.1,Stein..2019.all)

##   
## Pearson's product-moment correlation  
##   
## data: X.1 and Stein..2019.all  
## t = 1.642, df = 7, p-value = 0.1446  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.2105269 0.8824165  
## sample estimates:  
## cor   
## 0.5273227

abline(lm(Stein..2019.all~X.1),col="brown")



###Using Spearman's  
cor.test(X.1,Russell..2002,method="spearman")

## Warning in cor.test.default(X.1, Russell..2002, method = "spearman"): Cannot  
## compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and Russell..2002  
## S = 48.702, p-value = 0.09158  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.5941475

cor.test(X.1,White.et.al...1998,method="spearman")

## Warning in cor.test.default(X.1, White.et.al...1998, method = "spearman"):  
## Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and White.et.al...1998  
## S = 17.604, p-value = 0.01954  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.7904333

cor.test(X.1,Lehman..1987,method="spearman")

## Warning in cor.test.default(X.1, Lehman..1987, method = "spearman"): Cannot  
## compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and Lehman..1987  
## S = 2.5328, p-value = 0.007666  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.9276337

cor.test(X.1,Horner.and.Goodwin,method="spearman")

## Warning in cor.test.default(X.1, Horner.and.Goodwin, method = "spearman"):  
## Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and Horner.and.Goodwin  
## S = 10.593, p-value = 0.02692  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.8108437

cor.test(X.1,Lyson.and.Longrich..2011,method="spearman")

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and Lyson.and.Longrich..2011  
## S = 32, p-value = 0.03112  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.7333333

cor.test(X.1,Stein..2019.artic.assoc,method="spearman")

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and Stein..2019.artic.assoc  
## S = 12, p-value = 0.002028  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.9

cor.test(X.1,Stein..2019.all,method="spearman")

##   
## Spearman's rank correlation rho  
##   
## data: X.1 and Stein..2019.all  
## S = 54, p-value = 0.1328  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.55

cor.test(X.1[c(-8,-10)],Stein..2019.all[c(-8,-10)],method="spearman") #excluding Nodosauridae and Dromaeosauridae

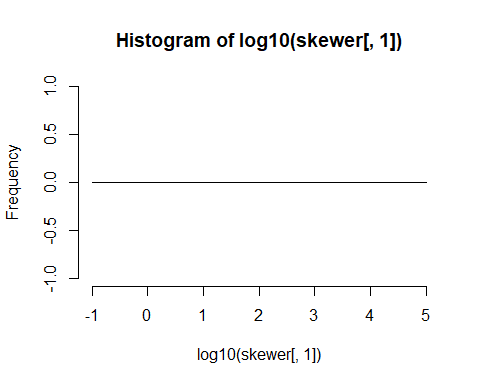
##   
## Spearman's rank correlation rho  
##   
## data: X.1[c(-8, -10)] and Stein..2019.all[c(-8, -10)]  
## S = 12, p-value = 0.01071  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.8571429

detach(hell)

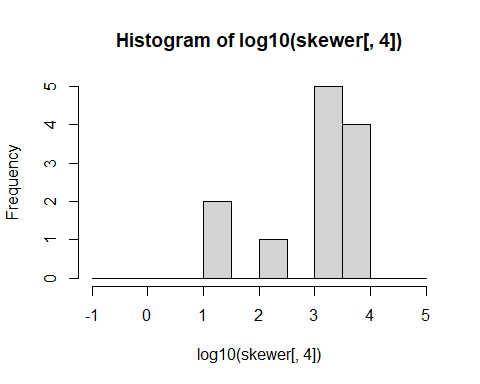
## Skew Analyses

Similar as above, data objects will need to be toggled on and off to access the different data types.

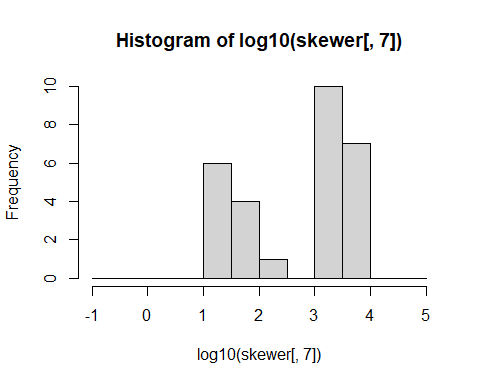
#from primary skew data  
hist(log10(skewer[,1]),breaks=c(seq(-1,5,1)))



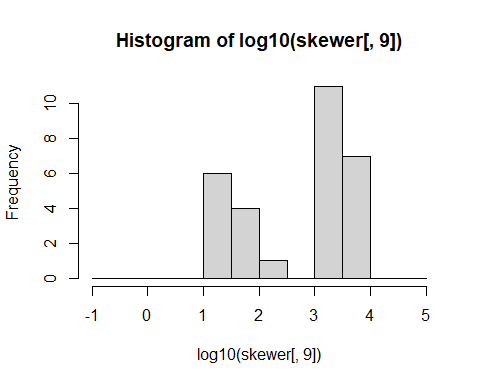
hist(log10(skewer[,4]),breaks=c(seq(-1,5,.5)))



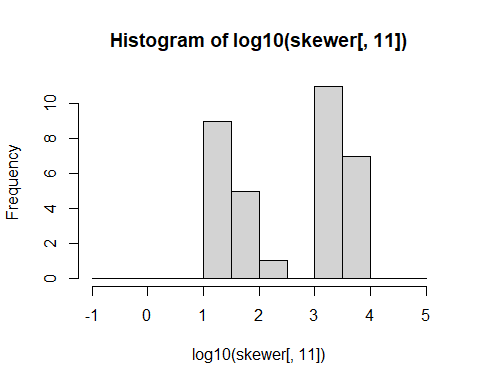
hist(log10(skewer[,7]),breaks=c(seq(-1,5,.5)))



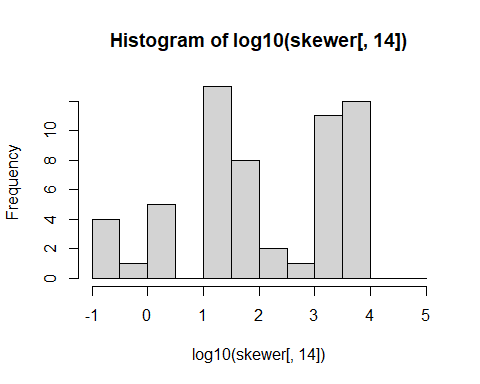
hist(log10(skewer[,9]),breaks=c(seq(-1,5,.5)))



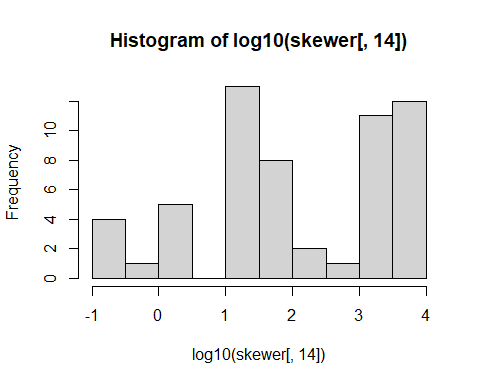
hist(log10(skewer[,11]),breaks=c(seq(-1,5,.5)))



hist(log10(skewer[,14]),breaks=c(seq(-1,5,.5)))



hist(log10(skewer[,14]))



skewness(log10(skewer[,1]),na.rm=T)

## [1] NaN

skewness(log10(skewer[,2]),na.rm=T)

## [1] NaN

skewness(log10(skewer[,3]),na.rm=T)

## [1] -0.4427287

skewness(log10(skewer[,4]),na.rm=T)

## [1] -1.095987

skewness(log10(skewer[,5]),na.rm=T)

## [1] -0.8773467

skewness(log10(skewer[,6]),na.rm=T)

## [1] -0.5878972

skewness(log10(skewer[,7]),na.rm=T)

## [1] -0.4931828

skewness(log10(skewer[,8]),na.rm=T)

## [1] -0.4931828

skewness(log10(skewer[,9]),na.rm=T)

## [1] -0.5476041

skewness(log10(skewer[,10]),na.rm=T)

## [1] -0.3892638

skewness(log10(skewer[,11]),na.rm=T)

## [1] -0.2560704

skewness(log10(skewer[,12]),na.rm=T)

## [1] -0.2294109

skewness(log10(skewer[,13]),na.rm=T)

## [1] -0.4521048

skewness(log10(skewer[,14]),na.rm=T)

## [1] -0.441933

#Skewer 2 data set  
plot(skewer2[,1],skewer2[,3])  
points(skewer2[,1],skewer2[,5],pch=15)  
time<-c(seq(1,14,1))  
cor.test(time,skewer2[,3],method="pearson")

##   
## Pearson's product-moment correlation  
##   
## data: time and skewer2[, 3]  
## t = 2.5109, df = 12, p-value = 0.02736  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.08176265 0.85212159  
## sample estimates:  
## cor   
## 0.5868823

cor.test(time,skewer2[,5],method="pearson")

##   
## Pearson's product-moment correlation  
##   
## data: time and skewer2[, 5]  
## t = 2.5909, df = 10, p-value = 0.02691  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.09408675 0.88556945  
## sample estimates:  
## cor   
## 0.6337671

cor.test(time[c(-1,-2)],skewer2[c(-1,-2),3],method="spearman")

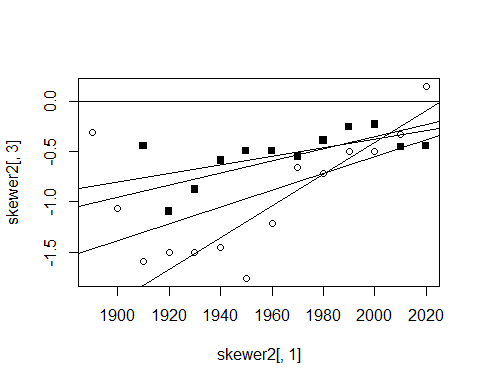
## Warning in cor.test.default(time[c(-1, -2)], skewer2[c(-1, -2), 3], method =  
## "spearman"): Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: time[c(-1, -2)] and skewer2[c(-1, -2), 3]  
## S = 23.079, p-value = 2.351e-05  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.9193039

cor.test(time[-3],skewer2[-3,5],method="pearson")

##   
## Pearson's product-moment correlation  
##   
## data: time[-3] and skewer2[-3, 5]  
## t = 3.8106, df = 9, p-value = 0.004149  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3515388 0.9417276  
## sample estimates:  
## cor   
## 0.7857214

abline(h=0)  
  
hell\_skew<-lm(skewer2[,3]~skewer2[,1])  
DPP\_skew<-lm(skewer2[,5]~skewer2[,1])  
abline(hell\_skew)  
abline(DPP\_skew)  
  
hell\_skew<-lm(skewer2[c(-1,-2),3]~skewer2[c(-1,-2),1])  
DPP\_skew<-lm(skewer2[-3,5]~skewer2[-3,1])  
abline(hell\_skew)  
abline(DPP\_skew)



# Mapping of Maastrichtian Formations

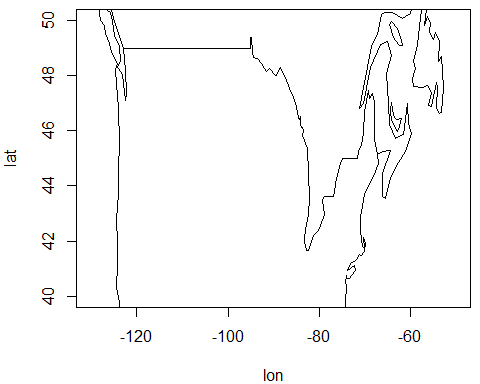
## Data sets

data(coastlineWorld)  
lon <- coastlineWorld[["longitude"]]  
lat <- coastlineWorld[["latitude"]]  
  
form <- Formations <- read.csv("Formations\_2.csv")

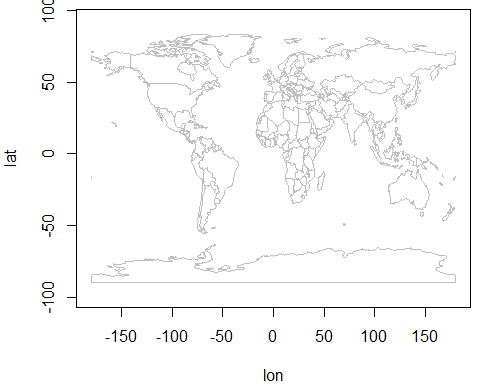
## Map plot

This code provides the raw plots used to generate the figure, but does not provide the final figure, which was generated by hand.

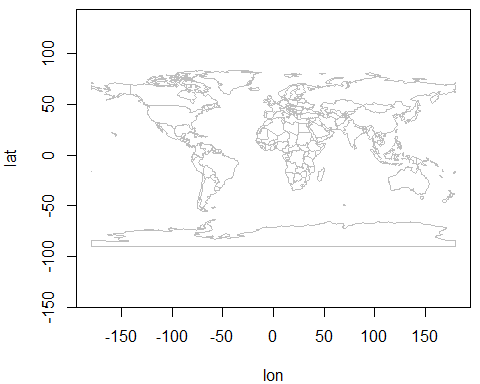
par(mar=c(4, 4, 0.5, 0.5))  
plot(lon, lat, type="l",  
 xlim=c(-130, -50), ylim=c(40, 50))



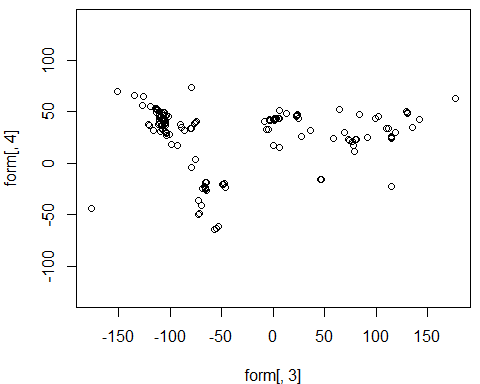
plot(lon, lat, type="l", asp=1/cos(45\*pi/180),col="grey")



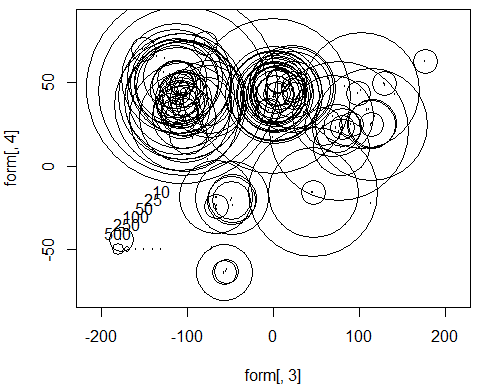
plot(lon, lat, type="l", asp=1,col="grey")



plot(form[,3],form[,4],asp=1)



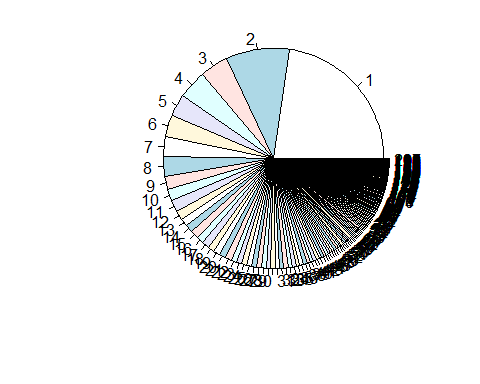
symbols(form[,3],form[,4],circles=log(form[,5]),inches=T)  
symbols(form[,3],form[,4],circles=(form[,6]/70),inches=F,add=T,lwd=1.5)  
symbols(-180,-50,circles=(500/70),inches=F,add=T,lwd=1.5)  
symbols(-170,-50,circles=(250/70),inches=F,add=T,lwd=1.5)  
symbols(-160,-50,circles=(100/70),inches=F,add=T,lwd=1.5)  
symbols(-150,-50,circles=(50/70),inches=F,add=T,lwd=1.5)  
symbols(-140,-50,circles=(25/70),inches=F,add=T,lwd=1.5)  
symbols(-130,-50,circles=(10/70),inches=F,add=T,lwd=1.5)  
text(-180,-40,labels="500")  
text(-170,-35,labels="250")  
text(-160,-30,labels="100")  
text(-150,-25,labels="50")  
text(-140,-20,labels="25")  
text(-130,-15,labels="10")



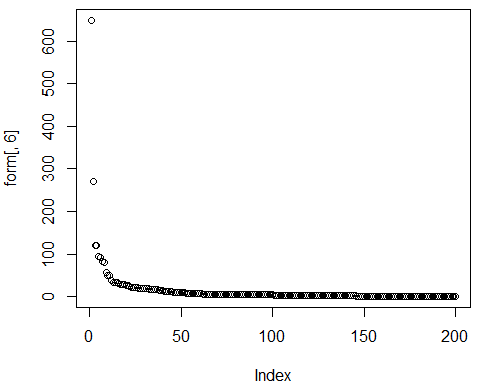
sort(form[,6])

## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [19] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [37] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [55] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [73] 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3  
## [91] 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4  
## [109] 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5  
## [127] 5 5 5 6 6 6 6 6 6 6 6 6 6 7 7 7 7 7  
## [145] 8 8 8 8 9 9 9 9 10 10 10 11 12 12 12 13 14 14  
## [163] 15 16 16 17 17 17 18 18 19 19 20 20 21 21 22 22 23 26  
## [181] 27 28 29 29 31 33 33 33 38 49 50 56 81 83 92 93 119 120  
## [199] 270 648

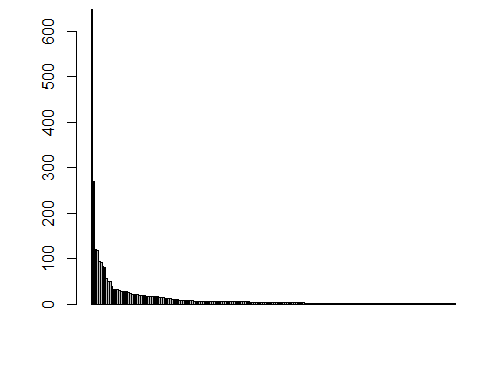
pie(form[,6])



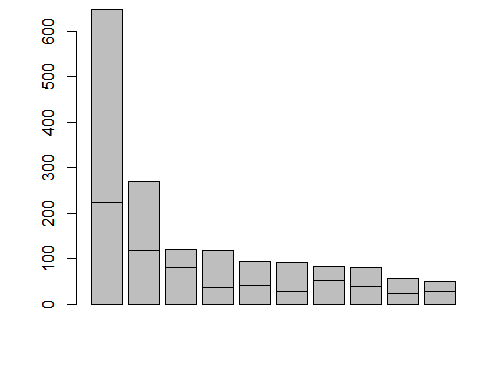
plot(form[,6])



barplot(form[,6])



barplot(form[c(1:10),6])  
barplot(form[c(1:10),5],add=T)



# Hell Creek Mammals

## Data sets

mam <- Mammal\_data\_5 <- read.csv("Mammal\_data\_5.csv")

## Mass vs. Completeness

#absolute mass  
cor.test(mam[,2],mam[,4],method = "pearson") #raw

##   
## Pearson's product-moment correlation  
##   
## data: mam[, 2] and mam[, 4]  
## t = 8.8084, df = 29, p-value = 1.081e-09  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.7150046 0.9272129  
## sample estimates:  
## cor   
## 0.853186

cor.test(log10(mam[,2]),mam[,4],method = "pearson") #log transformed

##   
## Pearson's product-moment correlation  
##   
## data: log10(mam[, 2]) and mam[, 4]  
## t = 2.7669, df = 29, p-value = 0.009749  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.1225101 0.6982746  
## sample estimates:  
## cor   
## 0.4570113

#rank mass  
cor.test(mam[,3],mam[,4],method = "spearman")

## Warning in cor.test.default(mam[, 3], mam[, 4], method = "spearman"): Cannot  
## compute exact p-value with ties

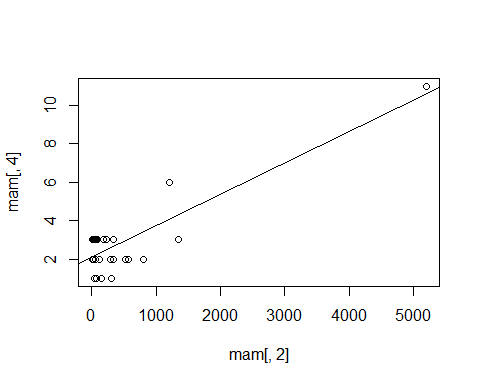
##   
## Spearman's rank correlation rho  
##   
## data: mam[, 3] and mam[, 4]  
## S = 4556.4, p-value = 0.6635  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.08136281

cor.test(mam[,3],mam[,4],method = "kendall")

## Warning in cor.test.default(mam[, 3], mam[, 4], method = "kendall"): Cannot  
## compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: mam[, 3] and mam[, 4]  
## z = 0.44228, p-value = 0.6583  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.06280872

#bivariate plot  
plot(mam[,2],mam[,4])  
abline(lm(mam[,4]~mam[,2]))



plot(log10(mam[,2]),mam[,4],col=mam[,5],xlim=c(0,4),pch=as.numeric(mam[,6]))

## Warning in plot.xy(xy, type, ...): NAs introduced by coercion

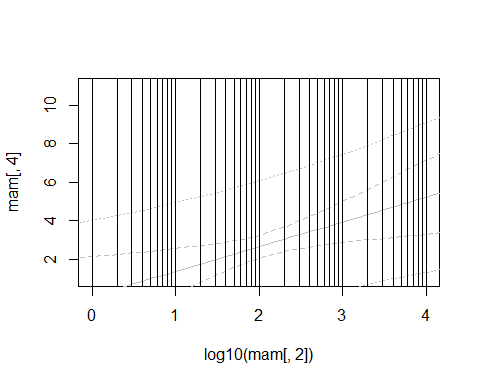
#bivariate model  
x <-log10(mam[,2])  
y <-mam[,4]  
xy <- data.frame(x, y)  
mdl <- lm(y ~ x, data = xy)  
predx <- data.frame(x = seq(from = -1, to = 5, by = 0.1))  
conf.int <- cbind(predx, predict(mdl, newdata = predx, interval = "confidence", level = 0.95))  
pred.int <- cbind(predx, predict(mdl, newdata = predx, interval = "prediction", level = 0.95))  
points(conf.int[,1],conf.int[,2],typ="l",col="grey")  
points(conf.int[,1],conf.int[,3],typ="l",lty=2,col="grey")  
points(conf.int[,1],conf.int[,4],typ="l",lty=2,col="grey")  
points(pred.int[,1],pred.int[,3],typ="l",lty=3,col="grey")  
points(pred.int[,1],pred.int[,4],typ="l",lty=3,col="grey")  
summary(mdl)

##   
## Call:  
## lm(formula = y ~ x, data = xy)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.2787 -1.2414 0.0183 0.7049 6.1447   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.06959 1.01032 0.069 0.94556   
## x 1.28787 0.46545 2.767 0.00975 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.651 on 29 degrees of freedom  
## Multiple R-squared: 0.2089, Adjusted R-squared: 0.1816   
## F-statistic: 7.656 on 1 and 29 DF, p-value: 0.009749

confint(mdl)

## 2.5 % 97.5 %  
## (Intercept) -1.9967356 2.135923  
## x 0.3359168 2.239814

#log lines  
abline(v=c(log10(seq(1,10,1))))  
abline(v=c(log10(seq(20,100,10))))  
abline(v=c(log10(seq(200,1000,100))))  
abline(v=c(log10(seq(2000,10000,1000))))



## Mass vs. Taphonomic mode

#absolute mass  
cor.test(mam[,2],mam[,5],method = "pearson") #raw

##   
## Pearson's product-moment correlation  
##   
## data: mam[, 2] and mam[, 5]  
## t = 5.0833, df = 29, p-value = 2.013e-05  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.4388228 0.8371494  
## sample estimates:  
## cor   
## 0.6864284

cor.test(mam[,2],mam[,5],method = "kendall") #raw

## Warning in cor.test.default(mam[, 2], mam[, 5], method = "kendall"): Cannot  
## compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: mam[, 2] and mam[, 5]  
## z = 0.81327, p-value = 0.4161  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.1200398

cor.test(log10(mam[,2]),mam[,5],method = "pearson") #log transformed

##   
## Pearson's product-moment correlation  
##   
## data: log10(mam[, 2]) and mam[, 5]  
## t = 2.1838, df = 29, p-value = 0.03721  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.02476156 0.64434222  
## sample estimates:  
## cor   
## 0.3758044

#rank mass  
cor.test(mam[,3],mam[,5],method = "spearman")

## Warning in cor.test.default(mam[, 3], mam[, 5], method = "spearman"): Cannot  
## compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: mam[, 3] and mam[, 5]  
## S = 4207.7, p-value = 0.4153  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## 0.151681

cor.test(mam[,3],mam[,5],method = "kendall")

## Warning in cor.test.default(mam[, 3], mam[, 5], method = "kendall"): Cannot  
## compute exact p-value with ties

##   
## Kendall's rank correlation tau  
##   
## data: mam[, 3] and mam[, 5]  
## z = 0.81327, p-value = 0.4161  
## alternative hypothesis: true tau is not equal to 0  
## sample estimates:  
## tau   
## 0.1200398