DATA 606 Group Project - Group 4

Zulihumaer Hailaiti, Maciej Pecak, Ewa Rambally, Zheyu Song, Hao Su

1. Introduction (Author: Ewa Rambally)

Our project for this class was based on a data set that accompanies a published article entitled: Body size trends in response to climate and urbanization in the widespread North American deer mouse, Peromyscus maniculatus. By:Guralnick, R.,Hantak, M., Li, D.,McLean, B., Publication date: May 21, 2020 Publisher: Dryad https://doi.org/10.5061/dryad.8w9ghx3j7 Data:https://orcid.org/0000-0001-9469-4741

1.1. Who is Deer Mouse? (Author: Ewa Rambally)

Deer mouse is a cousin of house mouse that is feared by epidemiologist and loved by pest-control companies and researchers in all sorts of biology fields. They usually weigh 10–24 g, and are 12–22 cm long from the tip of their little noses to the end of a slim tail. Their eyes pop out of the head surface and they are great at jumping. Unfortunately, they carry viruses and bacteria that are dangerous to humans. On the other hand, an almost complete omnipresence, adaptation to variety of environments, and quick reproduction process make them a perfect object of research and the "guinea pigs" of the labs. They are omnivores and like to eat insects, berries and nuts and at the same time often become a meal of smaller mammals, snake, and birds. A funny fact is that they display an OCD-like behavior by building unnecessarily oversized nests, especially in the lab environment. This data set comprised of many different data files. We considered the largest of them.

File: 13_pema_new_rezone_hblength_rezone_centroids28229obs1902_2017.csv

The data set of interest was collected and compiled and is available online with the access to the aforementioned paper. In its large version it measures 28,228 rows and 51 columns of both, categorical and numerical variables. The variable set includes: Data about mice: decade, decade2, source, zone, day, year, season, lifestage, sp (subspecies), sex, body_mass, tail_length, total_length, HB.length. Data about the environment the mouse lived in: Meteorological data: MAT, MWMT, MCMT, TD, MAP, MSP, AHM, SHM, DD_0, DD% DD_18, DD18, NFFD, cFFP, eFFP FFP, PAS, EMT, EXT, MAR, Eref, CMD, RH (these are standard abbreviations of weather and environmental conditions – see attachment) Data about the environment: long, lat, , pop_1_km2_log10, pop_4_km2_log10, pop_10km2_log10, ecoregion1, ecoregion2, ecoregion3, zone.longitude, zone.latitude.

1.2. Data cleaning and wrangling (Author: Ewa Rambally)

Data cleaning and initial preparation was conducted in Python.

File:Prep_for_kNNLifestage.ipynb

Many columns were dropped, and the final data set comprises of 18,358 rows and 35 columns out of which only selected were used for the final analysis.

After removing all the unspecified or non-numerical values in numerical columns (NaNs), there were only under 3000 observations left. Lots of information would have been unused if the data were not filled. For that purpose the kNN algorithm was used.

Most of the data were missing in the columns "lifestage" (about 25,000 missing values) and "body_mass"(about 19,000 missing values). In addition, presumable because of the compiling from multiple different sources for the period of more than a century, category names were not consistent i.e. some subspecies names were spelled in a few different ways, the category "ADULT" was represented by "Aduls, adult, ad, AD, mature, Mature"etc. The category names were unified for the categorical variables that were to be used in this projet, i.e. lifestage or species. The values for "lifestage" and "body_weight" were filled using the kNN algorithm.

PYTHON CODE: in the attachment. The resulting file: N5_18609r_CatUni___NaNinlfst.csv

1.2.1. kNN algorithm for categorical column "lifestage" (Author: Ewa Rambally)

Intake file: N5_18609r_CatUni___NaNinlfst.csv

For the purpose of best predictive properties of the kNN algorithm, to the categorical variable season, ecoregion1 and sex, assigned numerical values. For season and ecoregion1 the choice of numerical value of categories was somewhat ordering, to express distance and adjacency or its lack between category values in terms of season or physical location.

kNN was 90-95% accurate in predicting the ADULT category of the variable "lifestage", and 80-83% accurate in overall predictions (used 10-fold cross validation). (See: created data frame in the R code).

The resulting data frame is: mice_filled_lifestage_AD_SUBAD_YOUNG.csv.

This data frame was sent to Maciek Pecak for filling further the "body_mass" column.

R CODE:

```
library('ISLR')
library('ggplot2')
library("mlbench")
library("sampling")
library("MASS")
library("class") #for the knn() function
library("dplyr") #for select() function
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':
##
##     select

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("tidyr")
library("caret")
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'

## The following object is masked from 'package:sampling':
##
##     cluster
```

```
library("base")
library("lattice")
library("VGAM")
```

```
## Loading required package: stats4

## Loading required package: splines

##
## Attaching package: 'VGAM'

## The following object is masked from 'package:caret':
##
##     predictors
```

```
library("Rfast")
```

```
## Loading required package: Rcpp

## Loading required package: RcppZiggurat

##
## Attaching package: 'Rfast'

## The following object is masked from 'package:VGAM':
##
##     Rank

## The following object is masked from 'package:dplyr':
##
##     nth

## The following objects are masked from 'package:class':
##
##     knn, knn.cv
```

```
df=read.csv("N5_18609r_CatUni__NaNinlfst.csv")
```

```
unique(df$lifestage)
```

```
dim(df)
```

```
#removing remaining nans and "" and "U"
df=df[!is.na(df$pop_density_4km2),]
df=df[((df$ecoregion1 != "") & (df$lifestage != "FET") & (df$lifestage != "U")),]
unique(df$lifestage)
dim(df)
```

```
# coding reflect distance from ecoregion to ecoregion
u_num=c(7,6,6.5,7.5,1,2,4,3,5)#c(9,5,7,8,1,2,5,3,20,6)
u=unique(df$ecoregion1)
eco1_coded=data.frame(u,u_num)
#eco1_coded
```

```
# reflects distances from season taking into consideration difference in food supply (somewhat)
s_num=c(1,2,3,4)
s=unique(df$season)
seas_coded=data.frame(s,s_num)
#seas_coded
```

```
# "numerising needed categorical variables"
df$ecoregion1_num <- with(
  df,
  ifelse(ecoregion1==eco1_coded$u[1], eco1_coded$u_num[1],
         ifelse(ecoregion1==eco1_coded$u[2], eco1_coded$u_num[2],
                ifelse(ecoregion1==eco1_coded$u[3], eco1_coded$u_num[3],
                       ifelse(ecoregion1==eco1_coded$u[4], eco1_coded$u_num[4],
                              ifelse(ecoregion1==eco1_coded$u[5], eco1_coded$u_num[5],
                                     ifelse(ecoregion1==eco1_coded$u[6], eco1_coded$u_num[6],
                                            ifelse(ecoregion1==eco1_coded$u[7], eco1_coded$u_num[7],
                                                   ifelse(ecoregion1==eco1_coded$u[8], eco1_coded$u_num
                                                          ifelse(ecoregion1==eco1_coded$u[9], eco1_code
```

```
df$season_num <- with(
  df,
  ifelse(season==seas_coded$s[1], seas_coded$s_num[1],
         ifelse(season==seas_coded$s[2], seas_coded$s_num[2],
                ifelse(season==seas_coded$s[3], seas_coded$s_num[3],
                       ifelse(season==seas_coded$s[4], seas_coded$s_num[4], seas_coded$s_num[5] )))))
```

```
df$sex_num <- with(df,
                   ifelse(sex == "male", 1 ,0))
```

```
#normalize values in numerical variables
normalize <- function (x){
  return ((x-min(x))/(max(x)-min(x)))
}
df_select_norm<- as.data.frame(lapply(df_select, normalize))
```

```
#join with lifestage
df_select_norm$lifestage = df$lifestage
```

```
miss=df_select_norm[(df_select_norm$lifestage == ""),]
compl=df_select_norm[(df_select_norm$lifestage != ""),]
df_miss=df[df$lifestage == "",]
df_complete=df[df$lifestage != "",]
```

```
#CHECKING ACCURACY OF CATEGORY ASSIGNMENT
#create folds
set.seed(2023)
Folds10 <- createFolds(factor(compl$lifestage), k = 10, list=TRUE)
```

```
accuracy=c(1:10)
accuracy_AD=c(1:10)
accuracy_SUBAD=c(1:10)
accuracy_YOUNG=c(1:10)

find_accuracy <- function(x){sum(diag(x))/(sum(rowSums(x))) * 100}

for (i in 1:10){

  compl_factors_test=compl[unlist(Folds10[i]),]
  compl_target_category_test=compl_factors_test$lifestage
  compl_factors_test=compl_factors_test[-c(1,11)]

  compl_factors_train=compl[-unlist(Folds10[i]),]
  compl_target_category=compl_factors_train$lifestage
  compl_factors_train=compl_factors_train[-c(1,11)]

  pred_class <- knn(compl_factors_train,compl_factors_test,cl=compl_target_category,k=10)
  tab <- table(pred_class, compl_target_category_test)

  my_matrix <- matrix(tab, ncol=ncol(tab), dimnames=dimnames(tab))
  accuracy[i] <- find_accuracy(my_matrix)
  accuracy_AD[i] <- 100*tab[1]/(tab[1]+tab[2]+tab[3])
  accuracy_SUBAD[i] <- 100*tab[5]/(tab[4]+tab[5]+tab[6])
  accuracy_YOUNG[i] <- 100*tab[9]/(tab[7]+tab[8]+tab[9])
}

cv=mean(accuracy)
mean_accuracy_AD<-mean(accuracy_AD)
mean_accuracy_SUBAD<-mean(accuracy_SUBAD)
mean_accuracy_YOUNG<-mean(accuracy_YOUNG)

Accuracy_Rates = data.frame(
  c('AD+SUBAD+YOUNG', 'AD','SUBAD','YOUNG'),
  c(cv,mean_accuracy_AD, mean_accuracy_SUBAD,mean_accuracy_YOUNG)
)
colnames(Accuracy_Rates) = c('Group or Subgroup', 'Accuracy (%)')

Accuracy_Rates
#END OF CHECKING ACCURACY OF KNN ALGORITHM
```

```r
# APPLYING THE KNN ALGORITHM TO OUR DATA SET TO PERFORM THE FILL OF "lifestage" COLUMN
missing_factors_test <- miss[-c(1,11)]
complete_factors_traing <- compl[-c(1,11)]
complete_target <- compl$lifestage
missing_fill <-  knn(complete_factors_traing,missing_factors_test,cl=complete_target,k=10)
```

```r
df_miss$lifestage=missing_fill
final_df_fill_lifestage=rbind(df_complete,df_miss)
```

```r
#exporting filled data frame to .csv
library(readr)
write_csv(final_df_fill_lifestage,"mice_filled_lifestage_AD_SUBAD_YOUNG.csv")
```

Filling of the category "lifestage" was conducted on the data that contained the "body_mass" value. (18,609 observations). Afterwards, based on these values, the "body_mass" missing values were completed with the use of kNN algorithm (k=10)

1.2.2. kNN algorithm for numerical column "body_mass" (Author: Maciej Pecak)

The fill of "body_weight" was achieved also with the use of the kNN algorithm for finding 10 nearest neighbours and using bootstrap method to find their mean as the substitute for the missing value. The cross-validation error was about 40% ever after using the filled categories in the "lifestage" variable.

kNN implementation for filling the missing body mass was conducted in Python.

File: bodymass_fill.ipynb

---

2. Strata or Clusters (Author: Ewa Rambally)

For the purpose of the analysis and to see whether there is variation in mice's body characteristics (body mass, head-body length, and tail length) over time and in regards to climate, the time range 1929-2017 (was divided into three periods, "1926-1955", "1956-1985", "1986-2017", and the ecoregions level 1 into three climate groups. The column "climate_group" was created to clustered environments with similar properties ("FOREST" for lower land forests, westcoast forests, "GREAT PLAINS" for the central part of the US, and "DESERTS/DRY" for the remaining dry parts of the country).

The below code was conducted to determine whether the aforemantioned categories of each variable: "period" and "climate groups" were strata or clusters with regards to each of the variables" "body_mass", "tail_length", and "HB.Lenght" (head and bosy length: all but the tail). The code is provided for one of the pairs of the six pairs of data (period- tail_length):

- periods and body_mass
- periods and HB.Length
- periods and tail length
- ecoregions and body_mass
- ecoregions and HB.Length
- ecoregions and tail length

R CODE:

```
df_aov=read.csv("C:\\Users\\Testing5\\Documents\\EWA\\606\\606PROJECT\\Final_data\\mice_aov.csv")
```

Null hypothesis: the means of the different groups are the same Alternative hypothesis: At least one sample mean is not equal to the others.

```
#AoV tail_length vs. ecoregion
AOV=aov(tail_length~period, data=df_aov)
summary(AOV)
```

CONCLUSION: At least one sample mean is not equal to the others.

```
#Visualization
ggboxplot(df_aov, x="period", y="tail_length", color="period", xlab = "Time Period", ylab="tail_length"
```

For each pair test Ho: There are no difference in mean tail_length Ha: There is difference in mean tail_length

```
#Pairwise tests: Tukey Honest Significant Differences
TukeyHSD(AOV)
```

```
pairwise.t.test(df_aov$tail_length, df_aov$period,  p.adjust.method = "BH")
```

Conclusion: There are significant pairwise differences between tail length for each pair of the three periods.

Assumptions: 1. Homogeneity of variance (graph, Levene test). Relaxing homogeneity assumption - using Welsh one-way test (oneway.test(weight ~ group, data = my_data)) OR Pairwise t-tests with no assumption of equal variances pairwise.t.test(my_data$weight, my_data$group, p.adjust.method = "BH", pool.sd = FALSE) 2. Normality plot(bm_period_aov,2) and Shapiro-Wilk test

3. I assumptions are not met we can use Kruskal-Wallis rank sum test kruskal.test(weight ~ group, data = my_data)

ASSUMPTIONS:

1. HOMOGENEITY of Variance

```
plot((AOV), 1)
```

H_0: Variances of weights across periods are the same H_a: Varianes of weights across periods are different

```
leveneTest(tail_length~period, data=df_aov)
```

Reject the null hypothesis. Variances for different groups are different.

2. Normality:

```
plot(AOV, 2)
```

```r
# Extract the residuals
aov_residuals <- residuals(object=AOV )
# Run Shapiro-Wilk (3-5000 observations) test or Kolmogorov-Smirnoff test
ks.test(aov_residuals, "pnorm")
```

Tail_length is not normally distributed. Therefore, to check if mean body mass of the mice changed over time, we will use the Kruskal test.

```r
kruskal.test(tail_length ~ period, data = df_aov)
```

Conclusion: At least twao means of the tail lengths are no the same.

THEREFORE: Periods of time (as defined earlier) should be treated as STRATA for the variable tail_length .

The results of the above analysis and code results are summarized in the rable below:

Categories ->	Time Periods: (1) "1926-1955", (2) " 1956-1985", (3) "1986-2017"				
	SSB, SS	AoV Assumptions	Test for equality of means	Pairwise t-test (3 pairs)	Strata or Clusters
Body_mass	SSB=2206 SS=37 Potential STRATA	Homosc: No Norm: No	Kruskal-Wallis p-val<<0.01 Potential STRATA	All p-val <<0.01	EACH period is a STRATUM
HB.Length	SSB=3304 SS=120 Potential STRATA	Homosc: No Norm: No	Kruskal-Wallis p-val<<0.01 Potential STRATA	p-val << 0.01 except for pair (2)-(3) p-val= 0.38	STRATA: (1) (2)-(3)
Tail_length	SSB= 191052 SS=353 Potential STRATA	Homosc: No Norm: No	Kruskal-Wallis p-val=<<0.01 Potential STRATA	All p-val << 0.01	EACH period is a STRATUM

Figure 1: Figure 2.1 Analysis of the time periods

---

3. Population estimation (Author: Zulihumaer Halaiti)

Introduction

Our data set is collected from the research paper that are investigate on the environmental and urbanization effects on mice body size. So, our interested variable is body index of deer mice, like body mass and total length.

By doing sampling, we can have a general idea of population mean of body index. We will use body mass and tail length as an example.

Looking through all the variables that can be a candidate for strata or cluster, there aren't any variables suitable for cluster sampling. So, we will use simple random sampling and stratified sampling to do the population estimation and compare which method is better (more precise).

Categories	Ecoregions (9 groups)				
	SSB, SS	AoV Assumptions	AoV or Kruskal-Wallis test	Pairwise t-test (36 pairs)	Strata or Clusters
Body_mass	SSB=6024 SS=150 Potential STRATA	Homo: No Norm: No	Kruskal-Wallis p-val<<0.01 Potential STRATA	3 pairs p-val>0.1 Remaining pairs mostly with p-value <<0.01	Combining similar groups will lead to distinct STRATA
HB.Length	SSB=11470 SS=545 Potential STRATA	Homo: No Norm: No	Kruskal-Wallis p-val<<0.01 Potential STRATA	7 pairs with p-value >0.1 Remaining pairs mostly with p-value <<0.01	Combining similar groups will lead to distinct STRATA
Tail_length	SSB= 945363 SS=917 Potential STRATA	Homo: No Norm: No	Kruskal-Wallis p-val<<0.01 Potential STRATA	All p-val << 0.01 One pair with p-value <0.07	Each Ecoregion is a STRATUM

Figure 2: Figure 2.2 Analysis of the ecoregions

First, we checked if "Species" can be a candidate for strata by using several statistical method.

Original dataset

```
mice <- read.csv("mice_filled_all_values.csv")
mice <- mice[!mice$sp=='Peromyscus maniculatus',]
names(mice)
```

```
##  [1] "X.1"                  "X"                    "long"
##  [4] "lat"                  "decade"               "pop_density_4km2"
##  [7] "month"                "year"                 "season"
## [10] "lifestage"            "sp"                   "sex"
## [13] "body_mass"            "tail_length"          "total_length"
## [16] "HB.Length"            "MAT"                  "MWMT"
## [19] "MCMT"                 "TD"                   "MAP"
## [22] "MSP"                  "DD5"                  "FFP"
## [25] "EMT"                  "EXT"                  "ecoregion1"
## [28] "ecoregion1_num"       "season_num"           "sex_num"
## [31] "sex_transformed"      "ecoregion1_transformed" "season_transformed"
```

```
dim(mice)
```

```
## [1] 6737   33
```

```
unique(mice$sp)
```

```
##  [1] "Peromyscus maniculatus abietorum"    "Peromyscus maniculatus Wagner, 1845"
##  [3] "Peromyscus maniculatus sonoriensis"  "Peromyscus maniculatus bairdii"
##  [5] "Peromyscus maniculatus gambelii"     "Peromyscus maniculatus artemisiae"
##  [7] "Peromyscus maniculatus rufinus"      "Peromyscus maniculatus rubidus"
##  [9] "Peromyscus maniculatus austerus"     "Peromyscus maniculatus oreas"
## [11] "Peromyscus maniculatus nebrascensis" "Peromyscus maniculatus luteus"
```

```
## [13] "Peromyscus maniculatus gracilis"     "Peromyscus maniculatus blandus"
## [15] "Peromyscus maniculatus serratus"     "Peromyscus maniculatus osgoodi"
## [17] "Peromyscus maniculatus pallescens"   "Peromyscus maniculatus nubiterrae"
## [19] "Peromyscus maniculatus hollisteri"   "Peromyscus maniculatus santacruzae"
```

The mice species"Peromyscus maniculatus" was removed from data since we do not know the composition of this species. The data have 6737 rows and 33 columns.

Then we romove the sub species that have sample size less than 100, because the subspecies can not be sampled if its size is small

```
# Total observation of each sub-species
level=unique(mice$sp)
level
```

```
##  [1] "Peromyscus maniculatus abietorum"    "Peromyscus maniculatus Wagner, 1845"
##  [3] "Peromyscus maniculatus sonoriensis"  "Peromyscus maniculatus bairdii"
##  [5] "Peromyscus maniculatus gambelii"     "Peromyscus maniculatus artemisiae"
##  [7] "Peromyscus maniculatus rufinus"      "Peromyscus maniculatus rubidus"
##  [9] "Peromyscus maniculatus austerus"     "Peromyscus maniculatus oreas"
## [11] "Peromyscus maniculatus nebrascensis" "Peromyscus maniculatus luteus"
## [13] "Peromyscus maniculatus gracilis"     "Peromyscus maniculatus blandus"
## [15] "Peromyscus maniculatus serratus"     "Peromyscus maniculatus osgoodi"
## [17] "Peromyscus maniculatus pallescens"   "Peromyscus maniculatus nubiterrae"
## [19] "Peromyscus maniculatus hollisteri"   "Peromyscus maniculatus santacruzae"
```

```
list <-data.frame(table(mice$sp))
list
```

```
##                                  Var1 Freq
## 1     Peromyscus maniculatus abietorum  283
## 2    Peromyscus maniculatus artemisiae  193
## 3      Peromyscus maniculatus austerus   70
## 4       Peromyscus maniculatus bairdii  245
## 5       Peromyscus maniculatus blandus   13
## 6      Peromyscus maniculatus gambelii 1772
## 7      Peromyscus maniculatus gracilis   72
## 8    Peromyscus maniculatus hollisteri   20
## 9        Peromyscus maniculatus luteus  513
## 10 Peromyscus maniculatus nebrascensis  853
## 11   Peromyscus maniculatus nubiterrae   15
## 12        Peromyscus maniculatus oreas   10
## 13      Peromyscus maniculatus osgoodi   44
## 14   Peromyscus maniculatus pallescens    1
## 15      Peromyscus maniculatus rubidus  153
## 16      Peromyscus maniculatus rufinus  706
## 17  Peromyscus maniculatus santacruzae    8
## 18     Peromyscus maniculatus serratus   51
## 19  Peromyscus maniculatus sonoriensis 1098
## 20 Peromyscus maniculatus Wagner, 1845  617
```

```r
# The sub-species that has less than 100 observation was removed
df_filtered <- mice %>%
  group_by(sp) %>%
  filter(n() >= 100) %>%
  ungroup()
```

Statistical Check if "sp" can be a candidate for stratified sampling to estimate body mass

```r
# Perform ANOVA test
model <- lm(body_mass ~ sp, data = df_filtered)
anova_result <- anova(model)

# View the ANOVA table
anova_result
```

```
## Analysis of Variance Table
##
## Response: body_mass
##             Df Sum Sq Mean Sq F value    Pr(>F)
## sp           9   4461  495.71  29.683 < 2.2e-16 ***
## Residuals 6423 107265   16.70
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value for ANOVA table is < 0.05,it suggests that the differences between the groups are statistically significant, and that we can reject the null hypothesis that there are no differences between the groups.

To further explore which groups differ significantly from each other, so we use post-hoc tests to merge groups.

```r
library(agricolae)
CRD<-aov(body_mass~sp,data=df_filtered)
LS=LSD.test(CRD,trt="sp")
LS
```

```
## $statistics
##    MSerror   Df     Mean       CV
##   16.70019 6423 17.48655 23.36988
##
## $parameters
##         test p.ajusted name.t ntr alpha
##   Fisher-LSD      none     sp  10  0.05
##
## $means
##                                   body_mass      std    r      LCL      UCL
## Peromyscus maniculatus abietorum    18.42120 3.300447  283 17.94499 18.89741
## Peromyscus maniculatus artemisiae   17.96995 3.969825  193 17.39330 18.54660
## Peromyscus maniculatus bairdii      20.39388 4.571984  245 19.88207 20.90569
## Peromyscus maniculatus gambelii     16.72945 3.929728 1772 16.53914 16.91976
## Peromyscus maniculatus luteus       18.45653 3.360060  513 18.10283 18.81023
## Peromyscus maniculatus nebrascensis 17.16162 4.373857  853 16.88732 17.43591
## Peromyscus maniculatus rubidus      17.53203 3.253877  153 16.88437 18.17968
```

```
## Peromyscus maniculatus rufinus       18.10136 4.454749  706 17.79986 18.40286
## Peromyscus maniculatus sonoriensis   17.03461 4.147335 1098 16.79285 17.27637
## Peromyscus maniculatus Wagner, 1845  17.65883 4.444772  617 17.33632 17.98135
##                                       Min  Max  Q25  Q50  Q75
## Peromyscus maniculatus abietorum    11.7 32.8 16.1 18.4 20.4
## Peromyscus maniculatus artemisiae   11.2 35.5 14.9 17.7 21.0
## Peromyscus maniculatus bairdii      10.5 34.6 17.3 20.2 23.4
## Peromyscus maniculatus gambelii      9.2 34.0 14.0 16.5 19.0
## Peromyscus maniculatus luteus       10.0 33.5 16.0 18.0 20.5
## Peromyscus maniculatus nebrascensis  9.2 35.0 14.0 16.9 20.0
## Peromyscus maniculatus rubidus       9.1 26.9 15.2 17.1 19.6
## Peromyscus maniculatus rufinus       9.1 34.2 15.0 18.0 21.0
## Peromyscus maniculatus sonoriensis   9.3 35.0 14.0 17.0 19.0
## Peromyscus maniculatus Wagner, 1845  9.2 33.5 14.5 17.0 20.3
##
## $comparison
## NULL
##
## $groups
##                                     body_mass groups
## Peromyscus maniculatus bairdii       20.39388      a
## Peromyscus maniculatus luteus        18.45653      b
## Peromyscus maniculatus abietorum     18.42120      b
## Peromyscus maniculatus rufinus       18.10136      b
## Peromyscus maniculatus artemisiae    17.96995     bc
## Peromyscus maniculatus Wagner, 1845  17.65883      c
## Peromyscus maniculatus rubidus       17.53203     cd
## Peromyscus maniculatus nebrascensis  17.16162      d
## Peromyscus maniculatus sonoriensis   17.03461     de
## Peromyscus maniculatus gambelii      16.72945      e
##
## attr(,"class")
## [1] "group"
```

From the result above we will recreate new strata by merging the sub group that have no significant different means. And we have 5 stratas that are “a,b,c,d,e”(groupedspices)

```
# Assign the new strata to the merged sub-speices
df_filtered <- df_filtered%>%
  mutate(groupedspices = case_when(
    sp == "Peromyscus maniculatus bairdii" ~ "A",
    sp == "Peromyscus maniculatus luteus" ~ "B",
    sp == "Peromyscus maniculatus abietorum" ~ "B",
    sp == "Peromyscus maniculatus rufinus" ~ "B",
    sp == "Peromyscus maniculatus artemisiae" ~ "C",
    sp == "Peromyscus maniculatus Wagner, 1845" ~ "C",
    sp == "Peromyscus maniculatus rubidus" ~ "D",
    sp == "Peromyscus maniculatus nebrascensis" ~ "D",
    sp == "Peromyscus maniculatus sonoriensis" ~ "E",
    sp == "Peromyscus maniculatus gambelii" ~ "E",
    TRUE ~ "e"
  ))
```

Decide sample size

```
dim(df_filtered)
```

```
## [1] 6433   34
```

```
df_filtered
```

```
## # A tibble: 6,433 x 34
##      X.1     X   long   lat decade pop_densit~1 month  year season lifes~2 sp
##    <int> <int>  <dbl> <dbl>  <int>        <dbl> <int> <int> <chr>  <chr>   <chr>
##  1 18357 28224  -68.3  44.3   1950       4.32      11  1949 fall   YOUNG   Pero~
##  2 16217 24353 -118.   34.5   1970       0.0740    10  1972 fall   AD      Pero~
##  3 17431 26512 -119.   34.8   2000       0         11  2002 fall   AD      Pero~
##  4  2877 27705 -117.   47.7   1950       2.67       9  1948 fall   AD      Pero~
##  5 13965 20745 -120.   39.6   2000       0.776     10  2004 fall   AD      Pero~
##  6 13279 19702 -122.   38.3   2010     158.        12  2010 fall   AD      Pero~
##  7 13282 19716 -122.   38.3   2010     158.        11  2010 fall   AD      Pero~
##  8 13963 20741 -120.   39.6   2000       1.05      11  2003 fall   AD      Pero~
##  9 13966 20746 -120.   39.6   2000       1.05      11  2003 fall   AD      Pero~
## 10 15622 23259 -118.   34.5   1970       0.0740    10  1972 fall   AD      Pero~
## # ... with 6,423 more rows, 23 more variables: sex <chr>, body_mass <dbl>,
## #   tail_length <dbl>, total_length <dbl>, HB.Length <dbl>, MAT <dbl>,
## #   MWMT <dbl>, MCMT <dbl>, TD <dbl>, MAP <int>, MSP <int>, DD5 <int>,
## #   FFP <int>, EMT <dbl>, EXT <dbl>, ecoregion1 <chr>, ecoregion1_num <dbl>,
## #   season_num <int>, sex_num <int>, sex_transformed <int>,
## #   ecoregion1_transformed <dbl>, season_transformed <int>,
## #   groupedspices <chr>, and abbreviated variable names ...
```

```
# The formula is used to decide the sample size
z=1.96
p=0.5
e=0.05
samplesieze=ceiling((z^2*p*(1-p)/e^2)/(1+(z^2*p*(1-p)/(e^2*6433))))
samplesieze
```

```
## [1] 363
```

Then we use same sample size to run simple random sampling and stratified sampling

3.1 Simple random sampling

```
set.seed(10)
idx=sample(1:dim(df_filtered)[1],size=363,replace=FALSE)
mice1=df_filtered[idx,]
mice1=data.frame(mice1,pw=rep(dim(df_filtered)[1]/363,363),
                        fpc=rep(dim(df_filtered)[1],363))

svy1<-svydesign(id=~0, strata =NULL, weights=~pw, data = mice1, fpc=~fpc)
bodymassmean=svymean(~body_mass, svy1)
bodymassmean
```

```
##             mean     SE
## body_mass 17.674 0.2165
```

By doing simple random sampling, the estimated population mean of body mass is 17.674 and the standard devition is 0.2165.

3.2 Stratified sampling use proportional allocation principle

```
level=unique(df_filtered$groupedspices)
level
```

```
## [1] "B" "C" "E" "A" "D"
```

```
list <-data.frame(table(df_filtered$groupedspices))
list
```

```
##   Var1 Freq
## 1    A  245
## 2    B 1502
## 3    C  810
## 4    D 1006
## 5    E 2870
```

```
list <- list %>% mutate(size=round(list$Freq/(dim(df_filtered)[1]/363)))
colnames(list)[1]  <- "groupedspices"
list=list[c(2,3,5,1,4),]
list
```

```
##   groupedspices Freq size
## 2             B 1502   85
## 3             C  810   46
## 5             E 2870  162
## 1             A  245   14
## 4             D 1006   57
```

```
# Check if each strata have same probablity
set.seed(10)
idx<-sampling:::strata(df_filtered, stratanames=c("groupedspices"), size=
                         list$size, method="srswor")
micestrat<-getdata(df_filtered,idx)
```

```
micestrat=data.frame(micestrat, pw=1/micestrat$Prob,
fpc=c(rep(1502,85),rep(810,46),rep(2870,162),rep(245,14),rep(1006,57)))
svy2<-svydesign(id=~1,strata = ~sp, weights = ~pw, data = micestrat,
                fpc=~fpc)
mi2<-svymean(~body_mass, svy2)
mi2
```

```
##            mean     SE
## body_mass 17.23 0.2094
```

After running the simple random sampling and stratified sampling we can have a conclusion that for estimating body mass, stratified sampling is more precise with lower sample error that is 0.2094 while simple random sampling has sample error:0.2165.

3.3 We can also chek if "ecoregion" can be candidate for strata to estimate tail length

We remove the ecoregions that have observation less than 50, because the ecoregions can not be sampled if its size(primary sampling unit) is small

```
level1=unique(mice$ecoregion1)
level1
```

```
## [1] "EASTERN TEMPERATE FORESTS"       "MEDITERRANEAN CALIFORNIA"
## [3] "NORTHWESTERN FORESTED MOUNTAINS" "NORTH AMERICAN DESERTS"
## [5] "MARINE WEST COAST FOREST"        "GREAT PLAINS"
## [7] "NORTHERN FORESTS"                "TEMPERATE SIERRAS"
```

```
list1 <-data.frame(table(mice$ecoregion1))
list1
```

```
##                              Var1 Freq
## 1       EASTERN TEMPERATE FORESTS  357
## 2                    GREAT PLAINS  900
## 3        MARINE WEST COAST FOREST  151
## 4        MEDITERRANEAN CALIFORNIA  544
## 5          NORTH AMERICAN DESERTS 2571
## 6                NORTHERN FORESTS   51
## 7 NORTHWESTERN FORESTED MOUNTAINS 2152
## 8               TEMPERATE SIERRAS   11
```

```
mice1 <- mice %>%
  group_by(ecoregion1) %>%
  filter(n() >= 50) %>%
  ungroup()
```

Perform ANOVA test

```
model <- lm(tail_length ~ ecoregion1, data = mice1)
anova_result <- anova(model)

# View the ANOVA table
anova_result
```

```
## Analysis of Variance Table
##
## Response: tail_length
##              Df Sum Sq Mean Sq F value    Pr(>F)
## ecoregion1    6 274727   45788   617.5 < 2.2e-16 ***
```

```
## Residuals  6719 498214      74
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value for ANOVA table is < 0.05,it suggests that the differences between the groups are statistically significant, and that we can reject the null hypothesis that there are no differences between the groups.

To further explore which groups differ significantly from each other,which groups are no different in means. We use post-hoc tests to merge groups.

```
library(agricolae)
CRD<-aov(tail_length ~ ecoregion1,data=mice1)
LS=LSD.test(CRD,trt="ecoregion1")
LS
```

```
## $statistics
##    MSerror   Df    Mean       CV
##   74.15007 6719 68.2296 12.62069
##
## $parameters
##         test p.ajusted     name.t ntr alpha
##   Fisher-LSD      none ecoregion1   7  0.05
##
## $means
##                                 tail_length       std    r      LCL      UCL
## EASTERN TEMPERATE FORESTS          83.89636 11.291610  357 83.00295 84.78976
## GREAT PLAINS                       59.50000  7.684331  900 58.93732 60.06268
## MARINE WEST COAST FOREST           90.86424 10.171217  151 89.49053 92.23794
## MEDITERRANEAN CALIFORNIA           68.68419  7.183491  544 67.96045 69.40793
## NORTH AMERICAN DESERTS             65.79137  7.044791 2571 65.45845 66.12428
## NORTHERN FORESTS                   87.39216  8.191040   51 85.02843 89.75588
## NORTHWESTERN FORESTED MOUNTAINS    70.03717 10.234342 2152 69.67329 70.40106
##                                 Min Max Q25 Q50  Q75
## EASTERN TEMPERATE FORESTS        37 105  81  86 91.0
## GREAT PLAINS                     35  98  55  60 64.0
## MARINE WEST COAST FOREST         49 121  85  90 96.0
## MEDITERRANEAN CALIFORNIA         38 105  64  69 73.0
## NORTH AMERICAN DESERTS           35 106  61  66 70.0
## NORTHERN FORESTS                 74 107  81  86 94.5
## NORTHWESTERN FORESTED MOUNTAINS  35 116  64  69 75.0
##
## $comparison
## NULL
##
## $groups
##                                 tail_length groups
## MARINE WEST COAST FOREST           90.86424      a
## NORTHERN FORESTS                   87.39216      b
## EASTERN TEMPERATE FORESTS          83.89636      c
## NORTHWESTERN FORESTED MOUNTAINS    70.03717      d
## MEDITERRANEAN CALIFORNIA           68.68419      e
## NORTH AMERICAN DESERTS             65.79137      f
## GREAT PLAINS                       59.50000      g
##
```

```
## attr(,"class")
## [1] "group"
```

From the result above we can see all the ecoregion has different mean.

Decide sample size

```
dim(mice1)
```

```
## [1] 6726   33
```

```
# The formula is used to decide the sample size
z=1.96
p=0.5
e=0.05
samplesieze=ceiling((z^2*p*(1-p)/e^2)/(1+(z^2*p*(1-p)/(e^2*6626))))
samplesieze
```

```
## [1] 364
```

Then we use same sample size to run simple random sampling and stratified sampling

Simple random sampling

```
set.seed(10)
idx=sample(1:dim(mice1)[1],size=364,replace=FALSE)
mice2=mice1[idx,]
mice2=data.frame(mice2,pw=rep(dim(mice1)[1]/364,364),
                       fpc=rep(dim(mice1)[1],364))

svy1<-svydesign(id=~0, strata =NULL, weights=~pw, data = mice2, fpc=~fpc)
tailmean=svymean(~tail_length, svy1)
tailmean
```

```
##                mean     SE
## tail_length 68.216 0.5397
```

The estimated tail length by using simple random sampling is 67.81 and the standard devidation is 0.5424.

Stratified sampling use proportional allocation principle

```
level1=unique(mice1$ecoregion1)
level1
```

```
## [1] "EASTERN TEMPERATE FORESTS"       "MEDITERRANEAN CALIFORNIA"
## [3] "NORTHWESTERN FORESTED MOUNTAINS" "NORTH AMERICAN DESERTS"
## [5] "MARINE WEST COAST FOREST"        "GREAT PLAINS"
## [7] "NORTHERN FORESTS"
```

```
list1 <-data.frame(table(mice1$ecoregion1))
list1
```

```
##                              Var1 Freq
## 1       EASTERN TEMPERATE FORESTS  357
## 2                   GREAT PLAINS  900
## 3        MARINE WEST COAST FOREST  151
## 4        MEDITERRANEAN CALIFORNIA  544
## 5          NORTH AMERICAN DESERTS 2571
## 6                NORTHERN FORESTS   51
## 7 NORTHWESTERN FORESTED MOUNTAINS 2152
```

```
list1 <- list1 %>% mutate(size=round(list1$Freq/(dim(mice1)[1]/364)))
colnames(list1)[1]  <- "ecoregion1"
list1=list1[c(1,4,7,5,3,2,6),]
list1
```

```
##                         ecoregion1 Freq size
## 1       EASTERN TEMPERATE FORESTS  357   19
## 4        MEDITERRANEAN CALIFORNIA  544   29
## 7 NORTHWESTERN FORESTED MOUNTAINS 2152  116
## 5          NORTH AMERICAN DESERTS 2571  139
## 3        MARINE WEST COAST FOREST  151    8
## 2                   GREAT PLAINS  900   49
## 6                NORTHERN FORESTS   51    3
```

```
set.seed(10)
idx<-sampling:::strata(mice1, stratanames=c("ecoregion1"), size=
                       list1$size, method="srswor")
micestrat1<-getdata(mice1,idx)
```

```
micestrat1=data.frame(micestrat1, pw=1/micestrat1$Prob,
fpc=c(rep(357,19),rep(544,29),rep(2152,116),rep(2571,139),rep(151,8),rep(900,49),rep(51,3)))
svy2<-svydesign(id=~1,strata = ~ecoregion1, weights = ~pw, data = micestrat1,
                fpc=~fpc)
mi3<-svymean(~tail_length, svy2)
mi3
```

```
##                mean     SE
## tail_length 68.526 0.4754
```

After running the simple random sampling and stratified sampling we can have a conclusion that for estimating tail length, stratified sampling is more precise with lower sample error 0.4754 while sample error of simple random sampling is 0.5424.

---

4. Regression problems

This section will explore the regression problems - trying to predict mice body mass and the total length, based on the other available parameters. For each dependent variable there will be a linear regression model trained (with normality/homoscedasticity assuptions checked) as well as the regression tree.

4.1. Body mass - linear regression (Author: Maciej Pecak)

First, let's read the data.

```
mice.df <- read.csv("mice_filled_all_values.csv") %>%
  mutate(
    season = as.factor(season),
    lifestage = as.factor(lifestage),
    sex = as.factor(sex),
    ecoregion1 = as.factor(ecoregion1)
  )
head(mice.df, 3)
```

```
##     X.1     X       long      lat decade pop_density_4km2 month year season
## 1 18357 28224  -68.29874 44.33347   1950       4.31576252    11 1949   fall
## 2 16217 24353 -118.30000 34.53000   1970       0.07396096    10 1972   fall
## 3 17431 26512 -118.99695 34.78537   2000       0.00000000    11 2002   fall
##   lifestage                                  sp    sex body_mass tail_length
## 1     YOUNG    Peromyscus maniculatus abietorum   male      18.5          79
## 2        AD Peromyscus maniculatus Wagner, 1845   male      21.0          72
## 3        AD  Peromyscus maniculatus sonoriensis female      17.0          66
##   total_length HB.Length  MAT MWMT MCMT   TD  MAP MSP  DD5 FFP   EMT  EXT
## 1          151        72  8.4 21.2 -2.8 24.0 1136 374 2243 153 -32.7 44.1
## 2          164        92 15.7 26.0  6.5 19.5  131  18 3990 275 -10.3 43.7
## 3          155        89  8.9 18.4  1.2 17.2  411  15 1929 136 -24.5 37.5
##                  ecoregion1 ecoregion1_num season_num sex_num sex_transformed
## 1 EASTERN TEMPERATE FORESTS            1.0          3       1               1
## 2  MEDITERRANEAN CALIFORNIA            6.5          3       1               1
## 3  MEDITERRANEAN CALIFORNIA            6.5          3       0               0
##   ecoregion1_transformed season_transformed
## 1                      1                 2
## 2                      6                 2
## 3                      6                 2
```

Only mice subspecies with observation count greater than 100 will be considered.

```
species.considered <- c("Peromyscus maniculatus Wagner, 1845", "Peromyscus maniculatus sonoriensis", "P

#remove redundant/unused columns
model.df <- mice.df %>%
  select(-c(X.1, X, long, lat, decade, month, year, ecoregion1_num, season_num, sex_num, sex_transforme
  filter(sp %in% species.considered)
```

Next, the multicollinearity needs to be eliminated in order to have meaningful results.

```
vif(lm(body_mass ~
         pop_density_4km2 + tail_length + HB.Length + TD + MAP + MSP + FFP + EXT, data = model.df))
```

```
## pop_density_4km2      tail_length        HB.Length               TD
##         1.060649         1.368813         1.166767         2.424823
##              MAP              MSP              FFP              EXT
##         1.884722         2.405276         2.651601         3.369802
```

```
model.df3 <- model.df %>%
  select(-c(MCMT, MAT, DD5, EMT, MWMT, total_length))

vif(lm(body_mass ~ ., data = model.df3))
```

```
##                         GVIF Df GVIF^(1/(2*Df))
## pop_density_4km2    1.262984  1        1.123826
## season              2.346071  3        1.152719
## lifestage           1.840361  2        1.164731
## sp               1228.331676  8        1.559848
## sex                 1.026158  1        1.012995
## tail_length         1.906972  1        1.380932
## HB.Length           1.702573  1        1.304827
## TD                  3.939181  1        1.984737
## MAP                 3.252147  1        1.803371
## MSP                 4.912219  1        2.216353
## FFP                 5.094091  1        2.257009
## EXT                 9.915274  1        3.148853
## ecoregion1        575.890048  6        1.698354
```

Next, we create the first-order linear model and eliminate insignificant variables based on the p-value of the individual t-test.

```
base.model <- lm(body_mass ~ ., data = model.df3)
summary(base.model)
```

```
##
## Call:
## lm(formula = body_mass ~ ., data = model.df3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9055  -1.7326  -0.2077   1.4824  13.6678
##
## Coefficients:
##                                             Estimate Std. Error t value
## (Intercept)                                -1.010e+01  1.028e+00  -9.831
## pop_density_4km2                           -1.441e-04  2.654e-04  -0.543
## seasonspring                                1.268e+00  1.314e-01   9.649
## seasonsummer                                5.336e-01  1.248e-01   4.275
## seasonwinter                                9.557e-01  2.225e-01   4.294
## lifestageSUBAD                             -2.878e+00  1.070e-01 -26.904
## lifestageYOUNG                             -2.763e+00  1.195e-01 -23.129
## spPeromyscus maniculatus bairdii            4.350e+00  4.703e-01   9.251
## spPeromyscus maniculatus gambelii           1.362e+00  2.520e-01   5.405
## spPeromyscus maniculatus luteus            -5.049e-01  3.677e-01  -1.373
## spPeromyscus maniculatus nebrascensis       1.211e+00  2.617e-01   4.630
## spPeromyscus maniculatus rubidus            1.263e+00  7.881e-01   1.603
## spPeromyscus maniculatus rufinus            2.006e+00  2.525e-01   7.947
## spPeromyscus maniculatus sonoriensis        1.018e+00  2.538e-01   4.010
## spPeromyscus maniculatus Wagner, 1845       2.112e+00  2.776e-01   7.609
## sexmale                                    -5.658e-01  6.984e-02  -8.101
```

```
## tail_length                                  9.722e-02  5.354e-03  18.156
## HB.Length                                    2.351e-01  5.326e-03  44.135
## TD                                          -1.152e-02  1.332e-02  -0.866
## MAP                                         -1.651e-04  1.536e-04  -1.075
## MSP                                          1.519e-03  5.405e-04   2.810
## FFP                                         -8.366e-05  1.200e-03  -0.070
## EXT                                         -7.025e-02  1.912e-02  -3.675
## ecoregion1GREAT PLAINS                       4.461e+00  4.970e-01   8.977
## ecoregion1MARINE WEST COAST FOREST           1.907e+00  9.833e-01   1.939
## ecoregion1MEDITERRANEAN CALIFORNIA           3.383e+00  5.819e-01   5.814
## ecoregion1NORTH AMERICAN DESERTS             3.097e+00  5.503e-01   5.628
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS    2.960e+00  5.539e-01   5.344
## ecoregion1TEMPERATE SIERRAS                  2.860e+00  1.018e+00   2.810
##                                             Pr(>|t|)
## (Intercept)                                  < 2e-16 ***
## pop_density_4km2                             0.58733
## seasonspring                                 < 2e-16 ***
## seasonsummer                                1.94e-05 ***
## seasonwinter                                1.78e-05 ***
## lifestageSUBAD                               < 2e-16 ***
## lifestageYOUNG                               < 2e-16 ***
## spPeromyscus maniculatus bairdii             < 2e-16 ***
## spPeromyscus maniculatus gambelii           6.73e-08 ***
## spPeromyscus maniculatus luteus              0.16978
## spPeromyscus maniculatus nebrascensis       3.73e-06 ***
## spPeromyscus maniculatus rubidus             0.10895
## spPeromyscus maniculatus rufinus            2.25e-15 ***
## spPeromyscus maniculatus sonoriensis        6.16e-05 ***
## spPeromyscus maniculatus Wagner, 1845       3.17e-14 ***
## sexmale                                     6.55e-16 ***
## tail_length                                  < 2e-16 ***
## HB.Length                                    < 2e-16 ***
## TD                                           0.38679
## MAP                                          0.28239
## MSP                                          0.00497 **
## FFP                                          0.94441
## EXT                                          0.00024 ***
## ecoregion1GREAT PLAINS                       < 2e-16 ***
## ecoregion1MARINE WEST COAST FOREST           0.05253 .
## ecoregion1MEDITERRANEAN CALIFORNIA          6.40e-09 ***
## ecoregion1NORTH AMERICAN DESERTS            1.90e-08 ***
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS   9.45e-08 ***
## ecoregion1TEMPERATE SIERRAS                  0.00498 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.674 on 6121 degrees of freedom
## Multiple R-squared:  0.5961, Adjusted R-squared:  0.5942
## F-statistic: 322.6 on 28 and 6121 DF,  p-value: < 2.2e-16
```

```
sig.model.df <- model.df3 %>%
  select(-c(pop_density_4km2, TD, MAP, FFP))

sig.base.model <- lm(body_mass ~ ., sig.model.df)
```

```
summary(sig.base.model)
```

```
##
## Call:
## lm(formula = body_mass ~ ., data = sig.model.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9058  -1.7258  -0.2087   1.4828  13.6920
##
## Coefficients:
##                                          Estimate Std. Error t value
## (Intercept)                             -1.051e+01  9.480e-01 -11.090
## seasonspring                             1.263e+00  1.313e-01   9.618
## seasonsummer                             5.285e-01  1.236e-01   4.275
## seasonwinter                             9.589e-01  2.208e-01   4.344
## lifestageSUBAD                          -2.891e+00  1.065e-01 -27.139
## lifestageYOUNG                          -2.766e+00  1.192e-01 -23.204
## spPeromyscus maniculatus bairdii         4.355e+00  4.342e-01  10.030
## spPeromyscus maniculatus gambelii        1.395e+00  2.320e-01   6.015
## spPeromyscus maniculatus luteus         -4.888e-01  3.598e-01  -1.359
## spPeromyscus maniculatus nebrascensis    1.226e+00  2.581e-01   4.750
## spPeromyscus maniculatus rubidus         1.177e+00  7.544e-01   1.560
## spPeromyscus maniculatus rufinus         2.062e+00  2.430e-01   8.487
## spPeromyscus maniculatus sonoriensis     1.058e+00  2.441e-01   4.334
## spPeromyscus maniculatus Wagner, 1845    2.161e+00  2.590e-01   8.344
## sexmale                                 -5.653e-01  6.981e-02  -8.097
## tail_length                              9.703e-02  5.313e-03  18.263
## HB.Length                                2.352e-01  5.309e-03  44.289
## MSP                                      1.353e-03  5.225e-04   2.589
## EXT                                     -7.136e-02  1.318e-02  -5.414
## ecoregion1GREAT PLAINS                   4.532e+00  4.807e-01   9.428
## ecoregion1MARINE WEST COAST FOREST       2.017e+00  9.218e-01   2.188
## ecoregion1MEDITERRANEAN CALIFORNIA       3.510e+00  5.539e-01   6.337
## ecoregion1NORTH AMERICAN DESERTS         3.194e+00  5.261e-01   6.072
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS 3.022e+00  5.284e-01   5.719
## ecoregion1TEMPERATE SIERRAS              2.941e+00  1.001e+00   2.938
##                                         Pr(>|t|)
## (Intercept)                              < 2e-16 ***
## seasonspring                             < 2e-16 ***
## seasonsummer                            1.94e-05 ***
## seasonwinter                            1.42e-05 ***
## lifestageSUBAD                           < 2e-16 ***
## lifestageYOUNG                           < 2e-16 ***
## spPeromyscus maniculatus bairdii         < 2e-16 ***
## spPeromyscus maniculatus gambelii       1.90e-09 ***
## spPeromyscus maniculatus luteus          0.17435
## spPeromyscus maniculatus nebrascensis   2.08e-06 ***
## spPeromyscus maniculatus rubidus         0.11882
## spPeromyscus maniculatus rufinus         < 2e-16 ***
## spPeromyscus maniculatus sonoriensis    1.49e-05 ***
## spPeromyscus maniculatus Wagner, 1845    < 2e-16 ***
## sexmale                                 6.74e-16 ***
```

```
## tail_length                                   < 2e-16 ***
## HB.Length                                     < 2e-16 ***
## MSP                                           0.00966 **
## EXT                                          6.39e-08 ***
## ecoregion1GREAT PLAINS                        < 2e-16 ***
## ecoregion1MARINE WEST COAST FOREST            0.02868 *
## ecoregion1MEDITERRANEAN CALIFORNIA           2.51e-10 ***
## ecoregion1NORTH AMERICAN DESERTS             1.34e-09 ***
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS    1.12e-08 ***
## ecoregion1TEMPERATE SIERRAS                   0.00332 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.674 on 6125 degrees of freedom
## Multiple R-squared:  0.596,  Adjusted R-squared:  0.5944
## F-statistic: 376.4 on 24 and 6125 DF,  p-value: < 2.2e-16
```

After that, the interaction model has been trained and only significant interaction terms (as well as the base terms) were kept in the model.

```
int.bm.model <- lm(body_mass ~ (season + lifestage + sp + sex + tail_length + HB.Length + MSP + EXT + ec
                   , data = sig.model.df)

#summary(int.bm.model)
```

```
int.bm.model <- lm(body_mass ~ season + lifestage + sp + sex + tail_length + HB.Length + MSP + EXT + eco
                     season * lifestage + season * tail_length + season * HB.Length +
                     lifestage * sex + lifestage * HB.Length +
                     sp * sex + sp * tail_length + sp * MSP + sp * EXT +
                     sex * HB.Length + sex * MSP +
                     tail_length * EXT + tail_length * ecoregion1 +
                     HB.Length * MSP
                   , data = sig.model.df)

summary(int.bm.model)
```

```
##
## Call:
## lm(formula = body_mass ~ season + lifestage + sp + sex + tail_length +
##     HB.Length + MSP + EXT + ecoregion1 + season * lifestage +
##     season * tail_length + season * HB.Length + lifestage * sex +
##     lifestage * HB.Length + sp * sex + sp * tail_length + sp *
##     MSP + sp * EXT + sex * HB.Length + sex * MSP + tail_length *
##     EXT + tail_length * ecoregion1 + HB.Length * MSP, data = sig.model.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5263  -1.6558  -0.1696   1.3825  13.6666
##
## Coefficients:
##                                                    Estimate Std. Error
## (Intercept)                                      -8.999e+00  7.418e+00
## seasonspring                                     -2.669e+00  1.506e+00
```

```
## seasonsummer                                          2.687e+00  1.514e+00
## seasonwinter                                          4.069e+00  2.390e+00
## lifestageSUBAD                                        6.518e+00  1.321e+00
## lifestageYOUNG                                        1.900e+00  1.134e+00
## spPeromyscus maniculatus bairdii                     -1.732e+01  5.356e+00
## spPeromyscus maniculatus gambelii                    -1.634e-01  3.751e+00
## spPeromyscus maniculatus luteus                      -2.177e+01  5.292e+00
## spPeromyscus maniculatus nebrascensis                -7.160e+00  4.311e+00
## spPeromyscus maniculatus rubidus                      6.751e-01  7.399e+00
## spPeromyscus maniculatus rufinus                      1.344e+01  4.071e+00
## spPeromyscus maniculatus sonoriensis                 -5.198e+00  3.845e+00
## spPeromyscus maniculatus Wagner, 1845                -4.604e+00  4.054e+00
## sexmale                                               7.565e+00  9.693e-01
## tail_length                                           6.762e-02  9.305e-02
## HB.Length                                             2.981e-01  1.587e-02
## MSP                                                  -3.515e-03  3.418e-03
## EXT                                                   9.797e-03  1.616e-01
## ecoregion1GREAT PLAINS                               -3.584e+00  3.755e+00
## ecoregion1MARINE WEST COAST FOREST                   -1.313e+01  7.418e+00
## ecoregion1MEDITERRANEAN CALIFORNIA                   -1.018e+01  4.368e+00
## ecoregion1NORTH AMERICAN DESERTS                     -7.391e+00  4.112e+00
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS            -8.583e+00  4.105e+00
## ecoregion1TEMPERATE SIERRAS                          -5.897e+00  1.159e+01
## seasonspring:lifestageSUBAD                          -9.097e-01  4.036e-01
## seasonsummer:lifestageSUBAD                          -1.400e+00  3.409e-01
## seasonwinter:lifestageSUBAD                          -1.060e+00  1.112e+00
## seasonspring:lifestageYOUNG                          -3.090e-01  3.816e-01
## seasonsummer:lifestageYOUNG                          -6.861e-01  3.452e-01
## seasonwinter:lifestageYOUNG                          -9.695e-01  8.359e-01
## seasonspring:tail_length                              2.934e-02  1.341e-02
## seasonsummer:tail_length                             -2.084e-03  1.389e-02
## seasonwinter:tail_length                              4.120e-02  2.149e-02
## seasonspring:HB.Length                                2.522e-02  1.577e-02
## seasonsummer:HB.Length                               -2.007e-02  1.518e-02
## seasonwinter:HB.Length                               -6.442e-02  2.589e-02
## lifestageSUBAD:sexmale                                9.263e-01  2.025e-01
## lifestageYOUNG:sexmale                                1.504e-01  2.273e-01
## lifestageSUBAD:HB.Length                             -1.085e-01  1.592e-02
## lifestageYOUNG:HB.Length                             -5.668e-02  1.397e-02
## spPeromyscus maniculatus bairdii:sexmale             -1.177e+00  5.767e-01
## spPeromyscus maniculatus gambelii:sexmale            -1.155e+00  4.185e-01
## spPeromyscus maniculatus luteus:sexmale              -1.204e-01  4.642e-01
## spPeromyscus maniculatus nebrascensis:sexmale        -7.755e-01  4.223e-01
## spPeromyscus maniculatus rubidus:sexmale             -1.326e+00  5.646e-01
## spPeromyscus maniculatus rufinus:sexmale             -1.269e+00  4.250e-01
## spPeromyscus maniculatus sonoriensis:sexmale         -1.515e+00  4.263e-01
## spPeromyscus maniculatus Wagner, 1845:sexmale        -1.187e+00  4.638e-01
## spPeromyscus maniculatus bairdii:tail_length          1.713e-01  5.278e-02
## spPeromyscus maniculatus gambelii:tail_length        -6.747e-02  2.979e-02
## spPeromyscus maniculatus luteus:tail_length           8.438e-02  4.702e-02
## spPeromyscus maniculatus nebrascensis:tail_length     2.031e-02  3.354e-02
## spPeromyscus maniculatus rubidus:tail_length         -1.477e-01  7.625e-02
## spPeromyscus maniculatus rufinus:tail_length         -6.215e-02  3.258e-02
## spPeromyscus maniculatus sonoriensis:tail_length     -4.656e-03  3.180e-02
```

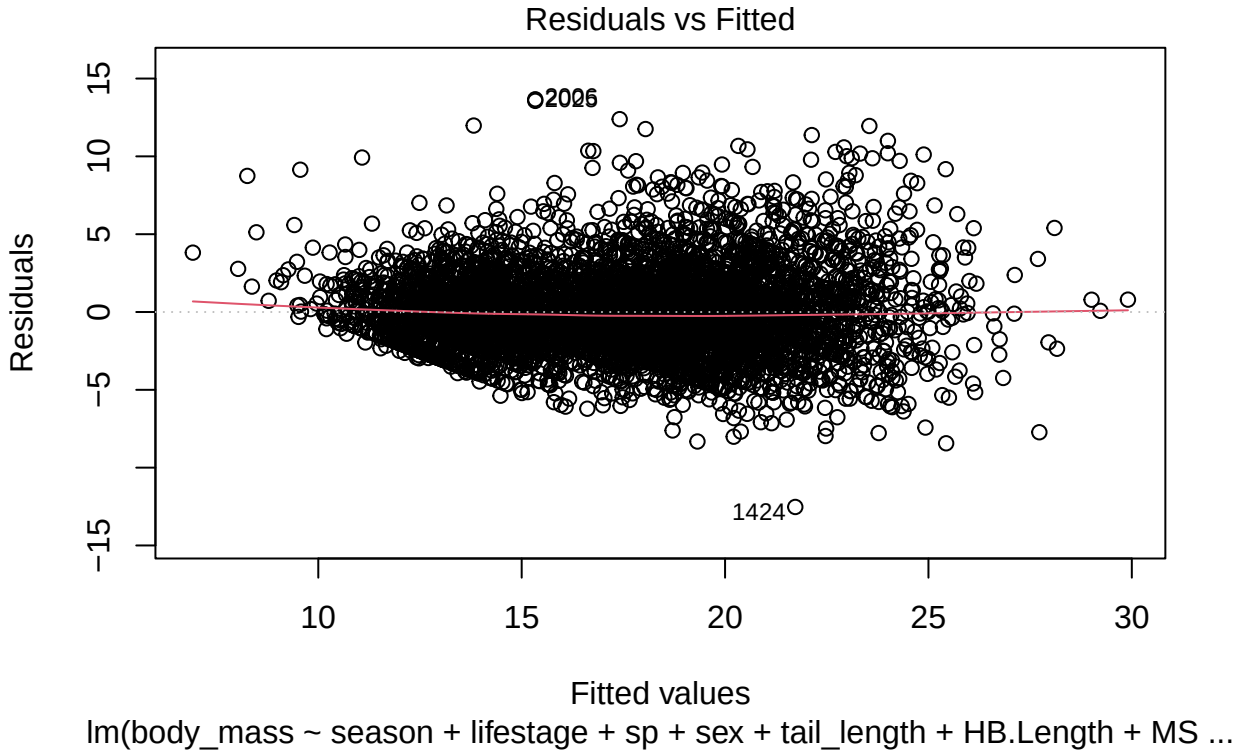
```
## spPeromyscus maniculatus Wagner, 1845:tail_length     -6.088e-02  3.247e-02
## spPeromyscus maniculatus bairdii:MSP                  -8.301e-04  2.459e-03
## spPeromyscus maniculatus gambelii:MSP                  4.985e-03  2.313e-03
## spPeromyscus maniculatus luteus:MSP                   -2.008e-02  4.375e-03
## spPeromyscus maniculatus nebrascensis:MSP              6.472e-03  2.374e-03
## spPeromyscus maniculatus rubidus:MSP                  -5.901e-03  3.660e-03
## spPeromyscus maniculatus rufinus:MSP                  -6.375e-03  2.377e-03
## spPeromyscus maniculatus sonoriensis:MSP               1.008e-02  2.409e-03
## spPeromyscus maniculatus Wagner, 1845:MSP              1.702e-03  2.654e-03
## spPeromyscus maniculatus bairdii:EXT                   2.961e-01  1.046e-01
## spPeromyscus maniculatus gambelii:EXT                  1.640e-01  7.726e-02
## spPeromyscus maniculatus luteus:EXT                    5.598e-01  1.067e-01
## spPeromyscus maniculatus nebrascensis:EXT              1.774e-01  8.905e-02
## spPeromyscus maniculatus rubidus:EXT                   3.312e-01  1.252e-01
## spPeromyscus maniculatus rufinus:EXT                  -1.535e-01  8.401e-02
## spPeromyscus maniculatus sonoriensis:EXT               1.672e-01  7.719e-02
## spPeromyscus maniculatus Wagner, 1845:EXT              2.879e-01  8.245e-02
## sexmale:HB.Length                                     -8.978e-02  9.937e-03
## sexmale:MSP                                            2.252e-03  9.174e-04
## tail_length:EXT                                       -3.334e-03  1.855e-03
## tail_length:ecoregion1GREAT PLAINS                     1.157e-01  5.898e-02
## tail_length:ecoregion1MARINE WEST COAST FOREST         2.494e-01  9.931e-02
## tail_length:ecoregion1MEDITERRANEAN CALIFORNIA         2.052e-01  6.850e-02
## tail_length:ecoregion1NORTH AMERICAN DESERTS           1.644e-01  6.505e-02
## tail_length:ecoregion1NORTHWESTERN FORESTED MOUNTAINS  1.780e-01  6.495e-02
## tail_length:ecoregion1TEMPERATE SIERRAS                1.489e-01  1.845e-01
## HB.Length:MSP                                          6.767e-06  3.156e-05
##                                                       t value Pr(>|t|)
## (Intercept)                                            -1.213 0.225130
## seasonspring                                           -1.772 0.076402 .
## seasonsummer                                            1.775 0.075957 .
## seasonwinter                                            1.703 0.088657 .
## lifestageSUBAD                                          4.936 8.19e-07 ***
## lifestageYOUNG                                          1.675 0.093902 .
## spPeromyscus maniculatus bairdii                       -3.233 0.001231 **
## spPeromyscus maniculatus gambelii                      -0.044 0.965255
## spPeromyscus maniculatus luteus                        -4.114 3.95e-05 ***
## spPeromyscus maniculatus nebrascensis                  -1.661 0.096774 .
## spPeromyscus maniculatus rubidus                        0.091 0.927302
## spPeromyscus maniculatus rufinus                        3.302 0.000966 ***
## spPeromyscus maniculatus sonoriensis                   -1.352 0.176437
## spPeromyscus maniculatus Wagner, 1845                  -1.135 0.256223
## sexmale                                                 7.805 6.97e-15 ***
## tail_length                                             0.727 0.467435
## HB.Length                                              18.784  < 2e-16 ***
## MSP                                                    -1.028 0.303930
## EXT                                                     0.061 0.951648
## ecoregion1GREAT PLAINS                                 -0.954 0.339936
## ecoregion1MARINE WEST COAST FOREST                     -1.770 0.076726 .
## ecoregion1MEDITERRANEAN CALIFORNIA                     -2.331 0.019800 *
## ecoregion1NORTH AMERICAN DESERTS                       -1.797 0.072331 .
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS              -2.091 0.036598 *
## ecoregion1TEMPERATE SIERRAS                            -0.509 0.610919
## seasonspring:lifestageSUBAD                            -2.254 0.024237 *
```

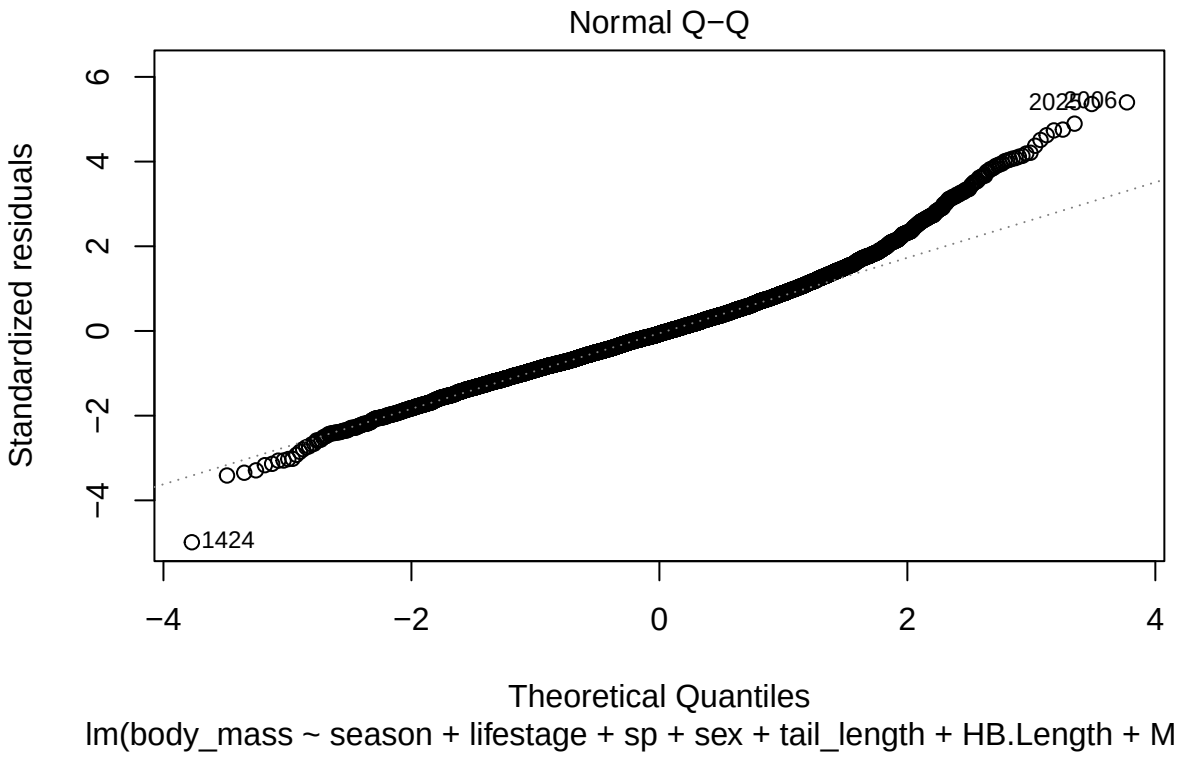
```
## seasonsummer:lifestageSUBAD                              -4.106 4.08e-05 ***
## seasonwinter:lifestageSUBAD                              -0.953 0.340570
## seasonspring:lifestageYOUNG                              -0.810 0.418088
## seasonsummer:lifestageYOUNG                              -1.988 0.046904 *
## seasonwinter:lifestageYOUNG                              -1.160 0.246163
## seasonspring:tail_length                                  2.189 0.028632 *
## seasonsummer:tail_length                                 -0.150 0.880715
## seasonwinter:tail_length                                  1.917 0.055297 .
## seasonspring:HB.Length                                    1.599 0.109813
## seasonsummer:HB.Length                                   -1.322 0.186153
## seasonwinter:HB.Length                                   -2.488 0.012865 *
## lifestageSUBAD:sexmale                                    4.575 4.86e-06 ***
## lifestageYOUNG:sexmale                                    0.662 0.508124
## lifestageSUBAD:HB.Length                                 -6.812 1.05e-11 ***
## lifestageYOUNG:HB.Length                                 -4.057 5.03e-05 ***
## spPeromyscus maniculatus bairdii:sexmale                 -2.040 0.041398 *
## spPeromyscus maniculatus gambelii:sexmale                -2.759 0.005815 **
## spPeromyscus maniculatus luteus:sexmale                  -0.259 0.795269
## spPeromyscus maniculatus nebrascensis:sexmale            -1.836 0.066363 .
## spPeromyscus maniculatus rubidus:sexmale                 -2.349 0.018857 *
## spPeromyscus maniculatus rufinus:sexmale                 -2.986 0.002836 **
## spPeromyscus maniculatus sonoriensis:sexmale             -3.552 0.000385 ***
## spPeromyscus maniculatus Wagner, 1845:sexmale            -2.559 0.010511 *
## spPeromyscus maniculatus bairdii:tail_length              3.246 0.001178 **
## spPeromyscus maniculatus gambelii:tail_length            -2.265 0.023568 *
## spPeromyscus maniculatus luteus:tail_length               1.795 0.072765 .
## spPeromyscus maniculatus nebrascensis:tail_length         0.606 0.544792
## spPeromyscus maniculatus rubidus:tail_length             -1.937 0.052848 .
## spPeromyscus maniculatus rufinus:tail_length             -1.907 0.056544 .
## spPeromyscus maniculatus sonoriensis:tail_length         -0.146 0.883581
## spPeromyscus maniculatus Wagner, 1845:tail_length        -1.875 0.060874 .
## spPeromyscus maniculatus bairdii:MSP                     -0.338 0.735744
## spPeromyscus maniculatus gambelii:MSP                     2.155 0.031181 *
## spPeromyscus maniculatus luteus:MSP                      -4.589 4.55e-06 ***
## spPeromyscus maniculatus nebrascensis:MSP                 2.726 0.006421 **
## spPeromyscus maniculatus rubidus:MSP                     -1.612 0.106952
## spPeromyscus maniculatus rufinus:MSP                     -2.682 0.007337 **
## spPeromyscus maniculatus sonoriensis:MSP                  4.185 2.90e-05 ***
## spPeromyscus maniculatus Wagner, 1845:MSP                 0.641 0.521390
## spPeromyscus maniculatus bairdii:EXT                      2.832 0.004647 **
## spPeromyscus maniculatus gambelii:EXT                     2.123 0.033834 *
## spPeromyscus maniculatus luteus:EXT                       5.246 1.61e-07 ***
## spPeromyscus maniculatus nebrascensis:EXT                 1.992 0.046405 *
## spPeromyscus maniculatus rubidus:EXT                      2.646 0.008177 **
## spPeromyscus maniculatus rufinus:EXT                     -1.827 0.067801 .
## spPeromyscus maniculatus sonoriensis:EXT                  2.167 0.030297 *
## spPeromyscus maniculatus Wagner, 1845:EXT                 3.492 0.000483 ***
## sexmale:HB.Length                                        -9.035  < 2e-16 ***
## sexmale:MSP                                               2.454 0.014138 *
## tail_length:EXT                                          -1.797 0.072442 .
## tail_length:ecoregion1GREAT PLAINS                        1.962 0.049831 *
## tail_length:ecoregion1MARINE WEST COAST FOREST            2.511 0.012063 *
## tail_length:ecoregion1MEDITERRANEAN CALIFORNIA            2.996 0.002748 **
## tail_length:ecoregion1NORTH AMERICAN DESERTS              2.528 0.011508 *
```

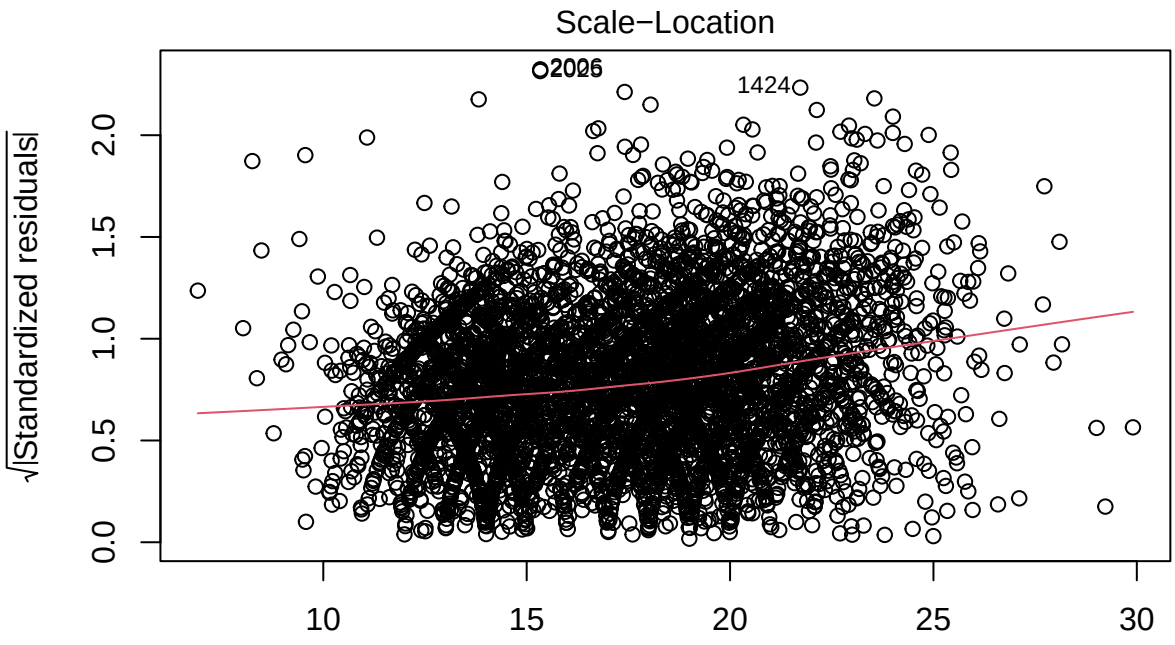
```
## tail_length:ecoregion1NORTHWESTERN FORESTED MOUNTAINS   2.741 0.006140 **
## tail_length:ecoregion1TEMPERATE SIERRAS                 0.807 0.419759
## HB.Length:MSP                                           0.214 0.830233
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.548 on 6067 degrees of freedom
## Multiple R-squared:  0.6366, Adjusted R-squared:  0.6317
## F-statistic: 129.6 on 82 and 6067 DF,  p-value: < 2.2e-16
```

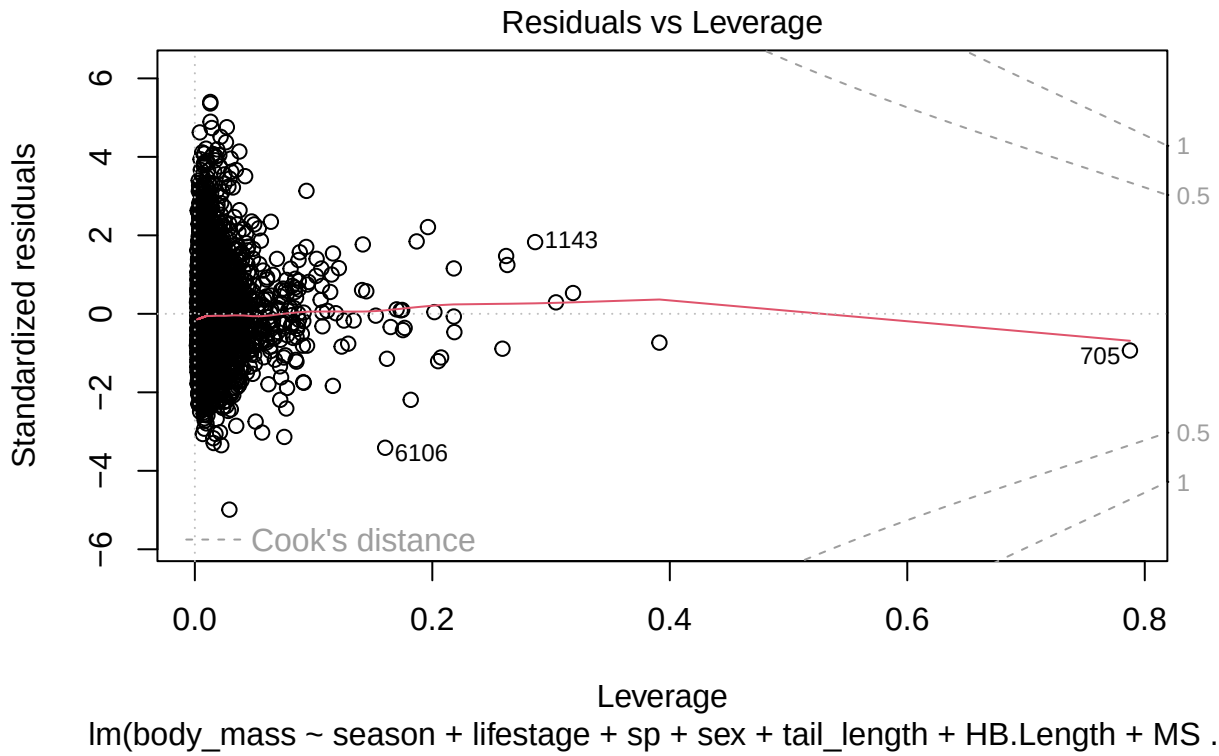
Finally, let's plot the useful graphs for the linear regression.

```
plot(int.bm.model)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(body_mass ~ season + lifestage + sp + sex + tail_length + HB.Length + MS ...
1424
2025
2006
−4
−2
0
2
4
6


Scale−Location
2026
1424
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
10
15
20
25
30
Fitted values
lm(body_mass ~ season + lifestage + sp + sex + tail_length + HB.Length + MS ...



Based on the above (graph 1) looks like we might have problem with homoscedasticity. Graph 2 also suggests that the might be no normality. Graph 4 shows that there are no significant outliers based on Cook's distance.

To check homoscedasticity, the Breusch-Pagan test has conducted.

```
library("lmtest")
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
##
## Attaching package: 'lmtest'
```

```
## The following object is masked from 'package:VGAM':
##
##     lrtest
```

```
bptest(int.bm.model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  int.bm.model
## BP = 486.08, df = 82, p-value < 2.2e-16
```

Based on the above p-value, we reject the null hypothesis, which means that the is no homoscedasticity.

In order to have comparison with other regression method, the cross validation error has been computed.

```
library("caret")
set.seed(10)
no.folds = 10
folds <- createFolds(model.df$sp, k = no.folds)
```

```
cv.errors <- numeric(no.folds)
for(i in 1:no.folds) {
  train <- model.df[-unlist(folds[i]), ]
  test <- model.df[unlist(folds[i]), ]

  bm.cv.regression <- lm(body_mass ~ season + lifestage + sp + sex + tail_length + HB.Length + MSP + EX
                      season * lifestage + season * tail_length + season * HB.Length +
                      lifestage * sex + lifestage * HB.Length +
                      sp * sex + sp * tail_length + sp * MSP + sp * EXT +
                      sex * HB.Length + sex * MSP +
                      tail_length * EXT + tail_length * ecoregion1 +
                      HB.Length * MSP
                    , data = train)

  predicted.vals <- predict(bm.cv.regression, test)


  cv.errors[i] <- sqrt(mean((test$body_mass - predicted.vals) ^ 2))
}

cv.errors
```

```
##  [1] 2.756484 2.577642 2.326079 2.443157 2.586758 2.543534 2.645546 2.656809
##  [9] 2.630121 2.514584
```

```
mean(cv.errors)
```

```
## [1] 2.568071
```

4.2.Total length - linear regression (Author: Maciej Pecak)

First, let's read the data.

```
mice.df <- read.csv("mice_filled_all_values.csv") %>%
  mutate(
    season = as.factor(season),
    lifestage = as.factor(lifestage),
    sex = as.factor(sex),
    ecoregion1 = as.factor(ecoregion1)
  )
```

Only mice subspecies with observation count greater than 100 will be considered.

```
species.considered <- c("Peromyscus maniculatus Wagner, 1845", "Peromyscus maniculatus sonoriensis", "Pe

model.df <- mice.df %>%
  select(-c(X.1, X, long, lat, decade, month, year, ecoregion1_num, season_num, sex_num, sex_transformed
  filter(sp %in% species.considered)
```

Next, the multicollinearity needs to be eliminated in order to have meaningful results.

```
vif(lm(total_length ~
         body_mass + pop_density_4km2 + tail_length + MAP + MSP + FFP + EXT + MCMT, data = model.df))
```

```
##        body_mass pop_density_4km2      tail_length              MAP
##         1.184046         1.055333         1.449087         1.620722
##              MSP              FFP              EXT             MCMT
##         2.733785         3.668097         2.922777         3.361104
```

```
model.df3 <- model.df %>%
  select(-c(TD, MAT, DD5, EMT, MWMT))

vif(lm(total_length ~ ., data = model.df3))
```

```
##                          GVIF Df GVIF^(1/(2*Df))
## pop_density_4km2    1.260712  1        1.122814
## season              2.374259  3        1.155016
## lifestage           2.119204  2        1.206544
## sp               1431.005101  8        1.574808
## sex                 1.037412  1        1.018534
## body_mass           2.475765  1        1.573456
## tail_length         2.010325  1        1.417859
## HB.Length           2.241647  1        1.497213
## MCMT                5.515000  1        2.348404
## MAP                 3.218546  1        1.794031
## MSP                 5.001953  1        2.236505
## FFP                 7.126438  1        2.669539
## EXT                 8.072399  1        2.841197
## ecoregion1        561.433110  6        1.694760
```

Next, we create the first-order linear model and eliminate insignificant variables based on the p-value of the individual t-test.

```
base.model <- lm(total_length ~ ., data = model.df3)
# summary(base.model)
```

```
sig.model.df <- model.df3 %>%
  select(-c(HB.Length, pop_density_4km2, EXT, FFP))

sig.base.model <- lm(total_length ~ ., sig.model.df)
# summary(sig.base.model)
```

After that, the interaction model has been trained and only significant interaction terms (as well as the base terms) were kept in the model.

```
int.len.model <- lm(total_length ~ (season + lifestage + sp + sex + tail_length + MCMT + MAP + MSP + eco
                    , data = sig.model.df)

# summary(int.len.model)
```

```
int.len.model <- lm(total_length ~ season + lifestage + sp + sex + tail_length + MCMT + MAP + MSP + eco
                      season * lifestage + season * tail_length + season * MAP + season * MSP +
                      lifestage * sex + lifestage * tail_length + lifestage * MAP + lifestage * MSP +
                      sp * sex + sp * tail_length + sp * MAP + sp * MSP +
                      sex * tail_length + sex * MSP +
                      tail_length * MSP +
                      MCMT * MAP + MCMT * ecoregion1 +
                      MAP * ecoregion1
                    , data = sig.model.df)

summary(int.len.model)
```

```
##
## Call:
## lm(formula = total_length ~ season + lifestage + sp + sex + tail_length +
##     MCMT + MAP + MSP + ecoregion1 + season * lifestage + season *
##     tail_length + season * MAP + season * MSP + lifestage * sex +
##     lifestage * tail_length + lifestage * MAP + lifestage * MSP +
##     sp * sex + sp * tail_length + sp * MAP + sp * MSP + sex *
##     tail_length + sex * MSP + tail_length * MSP + MCMT * MAP +
##     MCMT * ecoregion1 + MAP * ecoregion1, data = sig.model.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.356  -3.653   0.019   3.729  39.969
##
## Coefficients:
##                                                 Estimate Std. Error t value
## (Intercept)                                    1.083e+02  1.149e+01   9.422
## seasonspring                                  -1.533e+00  2.472e+00  -0.620
## seasonsummer                                   1.828e+01  2.487e+00   7.349
## seasonwinter                                   2.280e+00  3.944e+00   0.578
## lifestageSUBAD                                -1.652e-01  2.426e+00  -0.068
## lifestageYOUNG                                -9.360e+00  2.461e+00  -3.803
## spPeromyscus maniculatus bairdii               5.922e+00  6.942e+00   0.853
```

```
## spPeromyscus maniculatus gambelii                -1.261e+01  5.814e+00  -2.169
## spPeromyscus maniculatus luteus                   3.810e+01  7.057e+00   5.399
## spPeromyscus maniculatus nebrascensis            -2.299e-01  5.863e+00  -0.039
## spPeromyscus maniculatus rubidus                 -1.097e+01  9.422e+00  -1.165
## spPeromyscus maniculatus rufinus                 -6.887e+00  5.869e+00  -1.174
## spPeromyscus maniculatus sonoriensis             -1.023e+01  5.923e+00  -1.727
## spPeromyscus maniculatus Wagner, 1845            -2.276e+00  6.089e+00  -0.374
## sexmale                                           1.085e+01  2.169e+00   5.000
## tail_length                                       1.200e+00  7.765e-02  15.460
## MCMT                                              2.760e+00  5.577e-01   4.949
## MAP                                              -2.456e-02  9.489e-03  -2.588
## MSP                                              -4.108e-02  1.429e-02  -2.874
## ecoregion1GREAT PLAINS                           -4.534e+01  9.716e+00  -4.667
## ecoregion1MARINE WEST COAST FOREST               -2.383e+01  1.258e+01  -1.894
## ecoregion1MEDITERRANEAN CALIFORNIA               -3.153e+01  9.774e+00  -3.226
## ecoregion1NORTH AMERICAN DESERTS                 -3.710e+01  9.727e+00  -3.814
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS        -3.720e+01  9.725e+00  -3.825
## ecoregion1TEMPERATE SIERRAS                      -4.746e+01  1.426e+01  -3.327
## seasonspring:lifestageSUBAD                      -3.501e-01  9.833e-01  -0.356
## seasonsummer:lifestageSUBAD                      -3.943e+00  8.248e-01  -4.781
## seasonwinter:lifestageSUBAD                       1.092e+00  2.755e+00   0.396
## seasonspring:lifestageYOUNG                       2.112e+00  8.778e-01   2.405
## seasonsummer:lifestageYOUNG                      -4.903e+00  7.597e-01  -6.454
## seasonwinter:lifestageYOUNG                      -1.489e+00  1.999e+00  -0.745
## seasonspring:tail_length                          6.221e-02  3.585e-02   1.735
## seasonsummer:tail_length                         -2.115e-01  3.600e-02  -5.876
## seasonwinter:tail_length                          1.006e-02  5.657e-02   0.178
## seasonspring:MAP                                 -3.765e-04  9.101e-04  -0.414
## seasonsummer:MAP                                  3.418e-03  8.499e-04   4.021
## seasonwinter:MAP                                  1.585e-03  1.608e-03   0.986
## seasonspring:MSP                                 -2.751e-03  2.682e-03  -1.026
## seasonsummer:MSP                                 -8.758e-03  2.725e-03  -3.214
## seasonwinter:MSP                                 -6.270e-03  3.448e-03  -1.819
## lifestageSUBAD:sexmale                            2.647e+00  4.700e-01   5.631
## lifestageYOUNG:sexmale                            1.008e+00  4.949e-01   2.037
## lifestageSUBAD:tail_length                       -7.606e-02  3.268e-02  -2.327
## lifestageYOUNG:tail_length                        3.741e-02  3.518e-02   1.063
## lifestageSUBAD:MAP                                5.140e-04  6.162e-04   0.834
## lifestageYOUNG:MAP                               -1.300e-03  6.124e-04  -2.122
## lifestageSUBAD:MSP                               -6.511e-03  2.628e-03  -2.478
## lifestageYOUNG:MSP                               -3.168e-03  2.603e-03  -1.217
## spPeromyscus maniculatus bairdii:sexmale         -3.632e-01  1.451e+00  -0.250
## spPeromyscus maniculatus gambelii:sexmale        -4.185e+00  1.022e+00  -4.094
## spPeromyscus maniculatus luteus:sexmale          -2.847e+00  1.202e+00  -2.368
## spPeromyscus maniculatus nebrascensis:sexmale    -4.052e+00  1.072e+00  -3.779
## spPeromyscus maniculatus rubidus:sexmale          2.370e-03  1.402e+00   0.002
## spPeromyscus maniculatus rufinus:sexmale         -5.941e+00  1.064e+00  -5.584
## spPeromyscus maniculatus sonoriensis:sexmale     -4.802e+00  1.049e+00  -4.579
## spPeromyscus maniculatus Wagner, 1845:sexmale    -4.516e+00  1.133e+00  -3.985
## spPeromyscus maniculatus bairdii:tail_length     -1.138e-01  9.251e-02  -1.230
## spPeromyscus maniculatus gambelii:tail_length     1.512e-01  7.519e-02   2.011
## spPeromyscus maniculatus luteus:tail_length      -2.248e-01  8.309e-02  -2.705
## spPeromyscus maniculatus nebrascensis:tail_length 1.518e-02  7.647e-02   0.199
## spPeromyscus maniculatus rubidus:tail_length     -8.471e-02  8.970e-02  -0.944
```

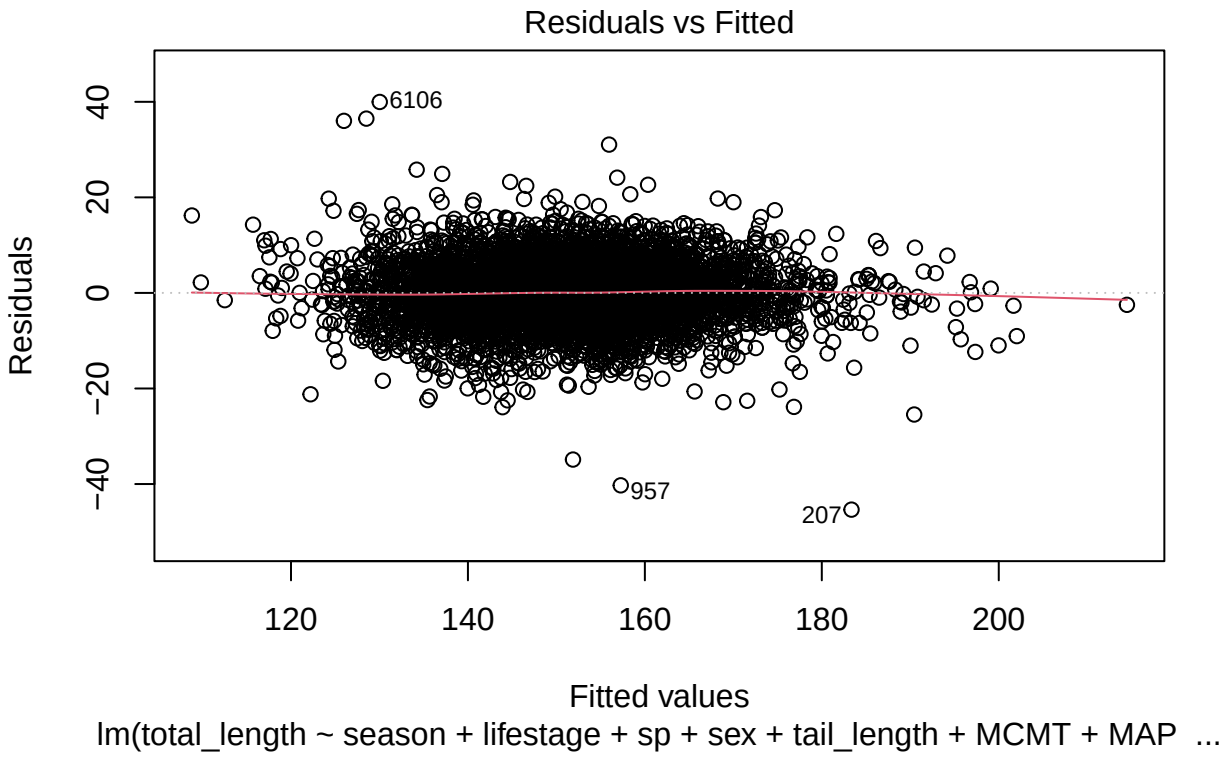
```
## spPeromyscus maniculatus rufinus:tail_length       1.321e-01  7.735e-02   1.708
## spPeromyscus maniculatus sonoriensis:tail_length   1.505e-01  7.624e-02   1.974
## spPeromyscus maniculatus Wagner, 1845:tail_length  1.805e-02  7.889e-02   0.229
## spPeromyscus maniculatus bairdii:MAP              -1.062e-02  4.294e-03  -2.473
## spPeromyscus maniculatus gambelii:MAP             -5.822e-03  2.564e-03  -2.271
## spPeromyscus maniculatus luteus:MAP               -1.177e-02  7.971e-03  -1.476
## spPeromyscus maniculatus nebrascensis:MAP          4.417e-03  4.954e-03   0.892
## spPeromyscus maniculatus rubidus:MAP               1.200e-02  7.615e-03   1.575
## spPeromyscus maniculatus rufinus:MAP              -1.030e-02  2.486e-03  -4.143
## spPeromyscus maniculatus sonoriensis:MAP          -1.378e-02  2.893e-03  -4.762
## spPeromyscus maniculatus Wagner, 1845:MAP         -1.970e-03  2.740e-03  -0.719
## spPeromyscus maniculatus bairdii:MSP               2.605e-02  8.958e-03   2.908
## spPeromyscus maniculatus gambelii:MSP              3.375e-02  8.342e-03   4.045
## spPeromyscus maniculatus luteus:MSP               -1.119e-02  1.614e-02  -0.694
## spPeromyscus maniculatus nebrascensis:MSP          1.666e-02  1.078e-02   1.546
## spPeromyscus maniculatus rubidus:MSP               2.477e-02  1.422e-02   1.742
## spPeromyscus maniculatus rufinus:MSP               4.457e-02  8.498e-03   5.245
## spPeromyscus maniculatus sonoriensis:MSP           5.294e-02  8.811e-03   6.008
## spPeromyscus maniculatus Wagner, 1845:MSP          1.836e-02  9.270e-03   1.981
## sexmale:tail_length                               -1.247e-01  2.423e-02  -5.147
## sexmale:MSP                                       -4.125e-03  2.212e-03  -1.864
## tail_length:MSP                                    3.830e-04  1.563e-04   2.450
## MCMT:MAP                                          -7.353e-05  9.919e-05  -0.741
## MCMT:ecoregion1GREAT PLAINS                       -2.976e+00  5.558e-01  -5.354
## MCMT:ecoregion1MARINE WEST COAST FOREST           -2.283e+00  5.751e-01  -3.970
## MCMT:ecoregion1MEDITERRANEAN CALIFORNIA           -3.042e+00  5.614e-01  -5.419
## MCMT:ecoregion1NORTH AMERICAN DESERTS             -2.867e+00  5.544e-01  -5.171
## MCMT:ecoregion1NORTHWESTERN FORESTED MOUNTAINS    -2.789e+00  5.512e-01  -5.060
## MCMT:ecoregion1TEMPERATE SIERRAS                  -1.608e+00  1.267e+00  -1.269
## MAP:ecoregion1GREAT PLAINS                         3.973e-02  8.964e-03   4.432
## MAP:ecoregion1MARINE WEST COAST FOREST             1.174e-02  1.159e-02   1.012
## MAP:ecoregion1MEDITERRANEAN CALIFORNIA             1.934e-02  9.327e-03   2.074
## MAP:ecoregion1NORTH AMERICAN DESERTS               2.877e-02  9.260e-03   3.106
## MAP:ecoregion1NORTHWESTERN FORESTED MOUNTAINS      2.776e-02  9.186e-03   3.022
## MAP:ecoregion1TEMPERATE SIERRAS                    3.747e-02  1.615e-02   2.320
##                                                  Pr(>|t|)
## (Intercept)                                       < 2e-16 ***
## seasonspring                                     0.535199
## seasonsummer                                     2.27e-13 ***
## seasonwinter                                     0.563249
## lifestageSUBAD                                   0.945716
## lifestageYOUNG                                   0.000144 ***
## spPeromyscus maniculatus bairdii                 0.393651
## spPeromyscus maniculatus gambelii                0.030120 *
## spPeromyscus maniculatus luteus                  6.95e-08 ***
## spPeromyscus maniculatus nebrascensis            0.968723
## spPeromyscus maniculatus rubidus                 0.244240
## spPeromyscus maniculatus rufinus                 0.240627
## spPeromyscus maniculatus sonoriensis             0.084285 .
## spPeromyscus maniculatus Wagner, 1845            0.708546
## sexmale                                          5.89e-07 ***
## tail_length                                       < 2e-16 ***
## MCMT                                             7.64e-07 ***
## MAP                                              0.009676 **
```

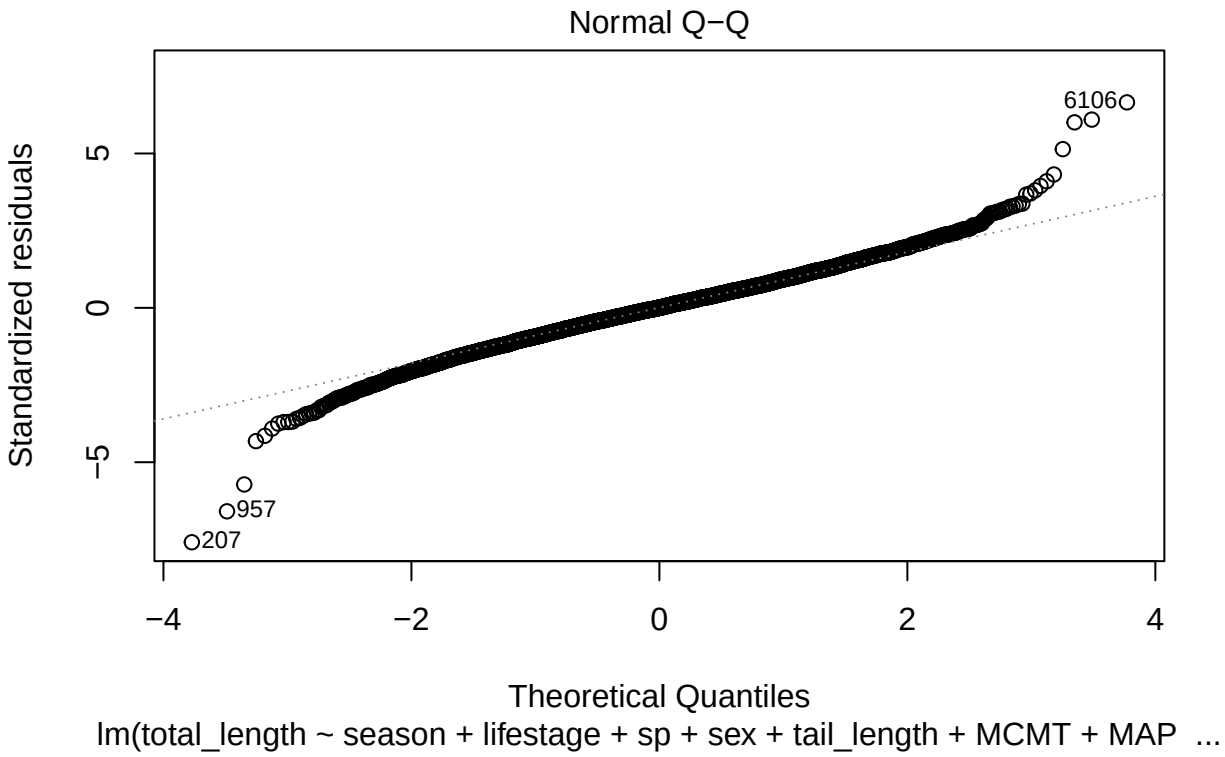
```
## MSP                                              0.004062 **
## ecoregion1GREAT PLAINS                           3.12e-06 ***
## ecoregion1MARINE WEST COAST FOREST               0.058257 .
## ecoregion1MEDITERRANEAN CALIFORNIA               0.001262 **
## ecoregion1NORTH AMERICAN DESERTS                 0.000138 ***
## ecoregion1NORTHWESTERN FORESTED MOUNTAINS        0.000132 ***
## ecoregion1TEMPERATE SIERRAS                      0.000883 ***
## seasonspring:lifestageSUBAD                      0.721860
## seasonsummer:lifestageSUBAD                      1.79e-06 ***
## seasonwinter:lifestageSUBAD                      0.691814
## seasonspring:lifestageYOUNG                      0.016182 *
## seasonsummer:lifestageYOUNG                      1.18e-10 ***
## seasonwinter:lifestageYOUNG                      0.456564
## seasonspring:tail_length                         0.082761 .
## seasonsummer:tail_length                         4.43e-09 ***
## seasonwinter:tail_length                         0.858849
## seasonspring:MAP                                 0.679125
## seasonsummer:MAP                                 5.86e-05 ***
## seasonwinter:MAP                                 0.324390
## seasonspring:MSP                                 0.304967
## seasonsummer:MSP                                 0.001316 **
## seasonwinter:MSP                                 0.069015 .
## lifestageSUBAD:sexmale                           1.87e-08 ***
## lifestageYOUNG:sexmale                           0.041703 *
## lifestageSUBAD:tail_length                       0.019977 *
## lifestageYOUNG:tail_length                       0.287749
## lifestageSUBAD:MAP                               0.404183
## lifestageYOUNG:MAP                               0.033859 *
## lifestageSUBAD:MSP                               0.013255 *
## lifestageYOUNG:MSP                               0.223507
## spPeromyscus maniculatus bairdii:sexmale         0.802285
## spPeromyscus maniculatus gambelii:sexmale        4.29e-05 ***
## spPeromyscus maniculatus luteus:sexmale          0.017912 *
## spPeromyscus maniculatus nebrascensis:sexmale    0.000159 ***
## spPeromyscus maniculatus rubidus:sexmale         0.998652
## spPeromyscus maniculatus rufinus:sexmale         2.45e-08 ***
## spPeromyscus maniculatus sonoriensis:sexmale     4.76e-06 ***
## spPeromyscus maniculatus Wagner, 1845:sexmale    6.83e-05 ***
## spPeromyscus maniculatus bairdii:tail_length     0.218911
## spPeromyscus maniculatus gambelii:tail_length    0.044373 *
## spPeromyscus maniculatus luteus:tail_length      0.006848 **
## spPeromyscus maniculatus nebrascensis:tail_length 0.842610
## spPeromyscus maniculatus rubidus:tail_length     0.345047
## spPeromyscus maniculatus rufinus:tail_length     0.087730 .
## spPeromyscus maniculatus sonoriensis:tail_length  0.048417 *
## spPeromyscus maniculatus Wagner, 1845:tail_length 0.818996
## spPeromyscus maniculatus bairdii:MAP             0.013436 *
## spPeromyscus maniculatus gambelii:MAP            0.023205 *
## spPeromyscus maniculatus luteus:MAP              0.139895
## spPeromyscus maniculatus nebrascensis:MAP        0.372625
## spPeromyscus maniculatus rubidus:MAP             0.115225
## spPeromyscus maniculatus rufinus:MAP             3.48e-05 ***
## spPeromyscus maniculatus sonoriensis:MAP         1.96e-06 ***
## spPeromyscus maniculatus Wagner, 1845:MAP        0.472047
```

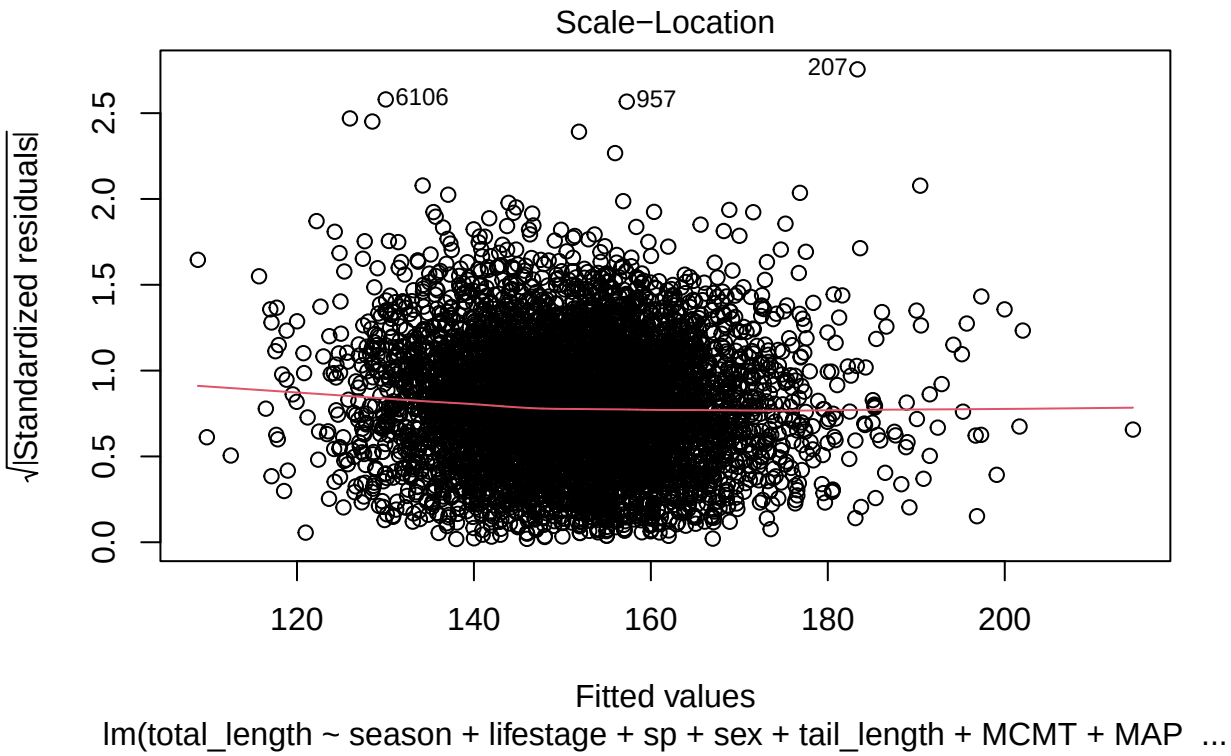
```
## spPeromyscus maniculatus bairdii:MSP              0.003648 **
## spPeromyscus maniculatus gambelii:MSP             5.29e-05 ***
## spPeromyscus maniculatus luteus:MSP               0.487993
## spPeromyscus maniculatus nebrascensis:MSP         0.122267
## spPeromyscus maniculatus rubidus:MSP              0.081534 .
## spPeromyscus maniculatus rufinus:MSP              1.62e-07 ***
## spPeromyscus maniculatus sonoriensis:MSP          1.99e-09 ***
## spPeromyscus maniculatus Wagner, 1845:MSP         0.047689 *
## sexmale:tail_length                               2.73e-07 ***
## sexmale:MSP                                       0.062315 .
## tail_length:MSP                                   0.014309 *
## MCMT:MAP                                          0.458508
## MCMT:ecoregion1GREAT PLAINS                       8.94e-08 ***
## MCMT:ecoregion1MARINE WEST COAST FOREST           7.27e-05 ***
## MCMT:ecoregion1MEDITERRANEAN CALIFORNIA           6.23e-08 ***
## MCMT:ecoregion1NORTH AMERICAN DESERTS             2.40e-07 ***
## MCMT:ecoregion1NORTHWESTERN FORESTED MOUNTAINS    4.32e-07 ***
## MCMT:ecoregion1TEMPERATE SIERRAS                  0.204622
## MAP:ecoregion1GREAT PLAINS                        9.50e-06 ***
## MAP:ecoregion1MARINE WEST COAST FOREST            0.311409
## MAP:ecoregion1MEDITERRANEAN CALIFORNIA            0.038138 *
## MAP:ecoregion1NORTH AMERICAN DESERTS              0.001903 **
## MAP:ecoregion1NORTHWESTERN FORESTED MOUNTAINS     0.002519 **
## MAP:ecoregion1TEMPERATE SIERRAS                   0.020372 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.132 on 6054 degrees of freedom
## Multiple R-squared:  0.7763, Adjusted R-squared:  0.7728
## F-statistic: 221.2 on 95 and 6054 DF,  p-value: < 2.2e-16
```

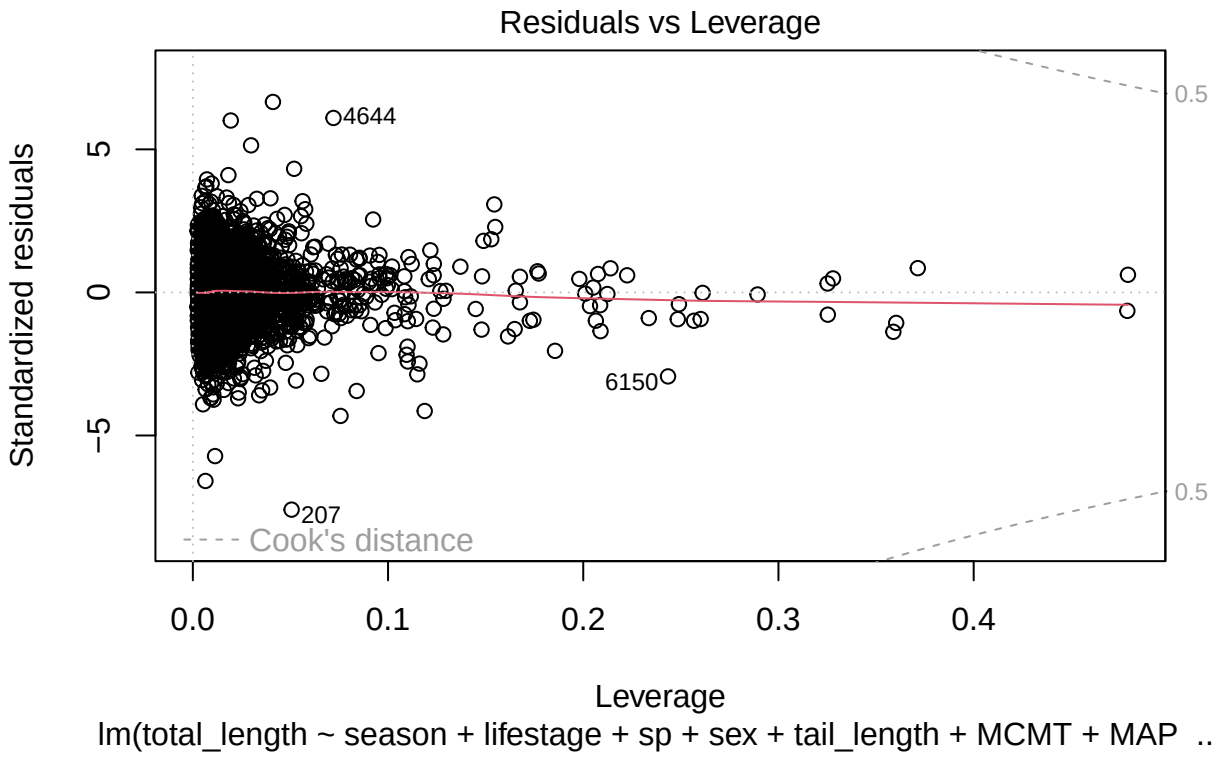
Finally, let's plot the useful graphs for the linear regression.

```
plot(int.len.model)
```


Residuals vs Fitted
Residuals
Fitted values
lm(total_length ~ season + lifestage + sp + sex + tail_length + MCMT + MAP ...
6106
957
207
40
20
0
−20
−40
120
140
160
180
200


Normal Q−Q
Standardized residuals
5
0
−5
6106
957
207
−4
−2
0
2
4
Theoretical Quantiles
lm(total_length ~ season + lifestage + sp + sex + tail_length + MCMT + MAP ...


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
6106
957
207
120
140
160
180
200
Fitted values
lm(total_length ~ season + lifestage + sp + sex + tail_length + MCMT + MAP ...



Based on the above (graph 1) looks like we might have problem with homoscedasticity. Graph 2 also suggests that the might be no normality. Graph 4 shows that there are no significant outliers based on Cook's distance.

To check homoscedasticity, the Breusch-Pagan test has conducted.

```
library("lmtest")
bptest(int.len.model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  int.len.model
## BP = 288.84, df = 95, p-value < 2.2e-16
```

Based on the above p-value, we reject the null hypothesis, which means that there is no homoscedasticity.

In order to have comparison with other regression method, the cross validation error has been computed.

```
library("caret")
set.seed(10)
no.folds = 5
folds <- createFolds(model.df$sp, k = no.folds)
```

```
cv.errors <- numeric(no.folds)
for(i in 1:no.folds) {
  train <- model.df[-unlist(folds[i]), ]
```

```
  test <- model.df[unlist(folds[i]), ]

  len.cv.regression <- lm(total_length ~ season + lifestage + sp + sex + tail_length + MCMT + MAP + MSP
                       season * lifestage + season * tail_length + season * MAP + season * MSP +
                       lifestage * sex + lifestage * tail_length + lifestage * MAP + lifestage * MSP +
                       sp * sex + sp * tail_length + sp * MAP + sp * MSP +
                       sex * tail_length + sex * MSP +
                       tail_length * MSP +
                       MCMT * MAP + MCMT * ecoregion1 +
                       MAP * ecoregion1
                     , data = train)

  predicted.vals <- predict(len.cv.regression, test)


  cv.errors[i] <- sqrt(sum((test$total_length - predicted.vals) ^ 2))
}

cv.errors
```

```
## [1] 212.3565 219.0424 217.0910 227.4575 214.8273
```

```
mean(cv.errors)
```

```
## [1] 218.1549
```

4.3. Regression tree - body mass (Author: Zheyu Song)

```
library(tree)
df<-read.csv("mice_filled_all_values.csv") %>% dplyr::select(-c(sex_transformed,ecoregion1_transformed,s
df <- df[df$sp != "Peromyscus maniculatus", ]
names(df)
```

```
##  [1] "pop_density_4km2" "season"           "lifestage"        "sp"
##  [5] "sex"              "body_mass"        "tail_length"      "total_length"
##  [9] "HB.Length"        "MAT"              "MWMT"             "MCMT"
## [13] "TD"               "MAP"              "MSP"              "DD5"
## [17] "FFP"              "EMT"              "EXT"              "ecoregion1"
```

```
dim(df)
```

```
## [1] 6737   20
```

```
# Convert variables to factors
df$ecoregion1 <- as.factor(df$ecoregion1)
df$sp <- as.factor(df$sp)
df$sex <- as.factor(df$sex)
df$lifestage <- as.factor(df$lifestage)
df$season <- as.factor(df$season)
```

```
# Remove redundant or not meaningful levels
df$ecoregion1 <- droplevels(df$ecoregion1)
df$sp <- droplevels(df$sp)
df$sex <- droplevels(df$sex)
df$lifestage <- droplevels(df$lifestage)
df$season <- droplevels(df$season)
```

Columns with irrelevant data were removed. The columns removed included "sex_transformed," "ecoregion1_transformed," "season_transformed," "sex_num," "season_num," "ecoregion1_num," "long," "lat," "decade," "month," "year," "X," and "X.1."

All entries with species "Peromyscus maniculatus" were removed from the dataset.

The variable types were converted to factors to prepare for modeling. The variables that were converted to factors included "ecoregion1," "sp," "sex," "lifestage," and "season."

Redundant or not meaningful levels were removed from the factors using the "droplevels" function.

The dataset was split into a training set and a test set using a random sample of 75% of the data for training and 25% for testing.

```
set.seed (10)


splitIndex <- sample(1:nrow(df), size = 3/4 * nrow(df))
train <- df[splitIndex, ]
test <- df[-splitIndex, ]
#head(train)
dim(test)
```

```
## [1] 1685   20
```

```
dim(train)
```

```
## [1] 5052   20
```

```
names(test)
```

```
##  [1] "pop_density_4km2" "season"           "lifestage"        "sp"
##  [5] "sex"              "body_mass"        "tail_length"      "total_length"
##  [9] "HB.Length"        "MAT"              "MWMT"             "MCMT"
## [13] "TD"               "MAP"              "MSP"              "DD5"
## [17] "FFP"              "EMT"              "EXT"              "ecoregion1"
```

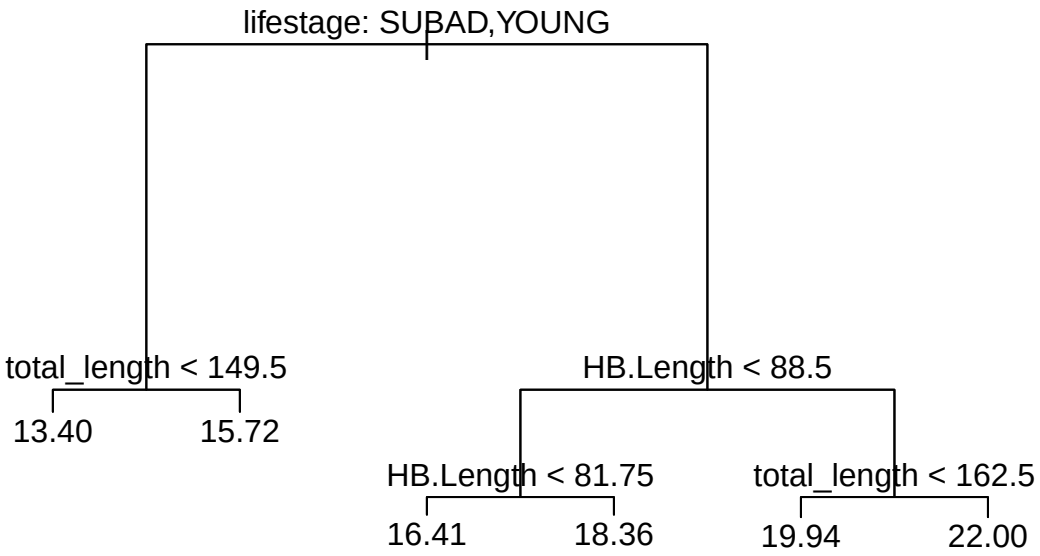
4.4. Regression Tree Model for Predicting Body Mass (Author: Zheyu Song)

A regression tree was built using the body mass as the target variable and the remaining variables as predictors. The tree was built using the tree function from the tree library. The model was summarized using the summary function, and a plot of the tree was produced using the plot function.

```
tree.mass<-tree(body_mass ~. , data=train)
summary(tree.mass)
```

```
##
## Regression tree:
## tree(formula = body_mass ~ ., data = train)
## Variables actually used in tree construction:
## [1] "lifestage"    "total_length" "HB.Length"
## Number of terminal nodes:  6
## Residual mean deviance:  8.428 = 42530 / 5046
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -10.7400  -2.0010  -0.3551   0.0000   1.5990  14.2600
```

```
plot(tree.mass)
text(tree.mass ,pretty =0)
```



The plot of the tree indicates the variables used in each split and the predicted body mass for each terminal node.

The regression tree had a total of 6 terminal nodes and a residual mean deviance of 8.428. The variables used in the construction of the tree were “lifestage”, “total_length”, and “HB.Length”. The tree had two levels of splits, with the first split being based on “lifestage”.

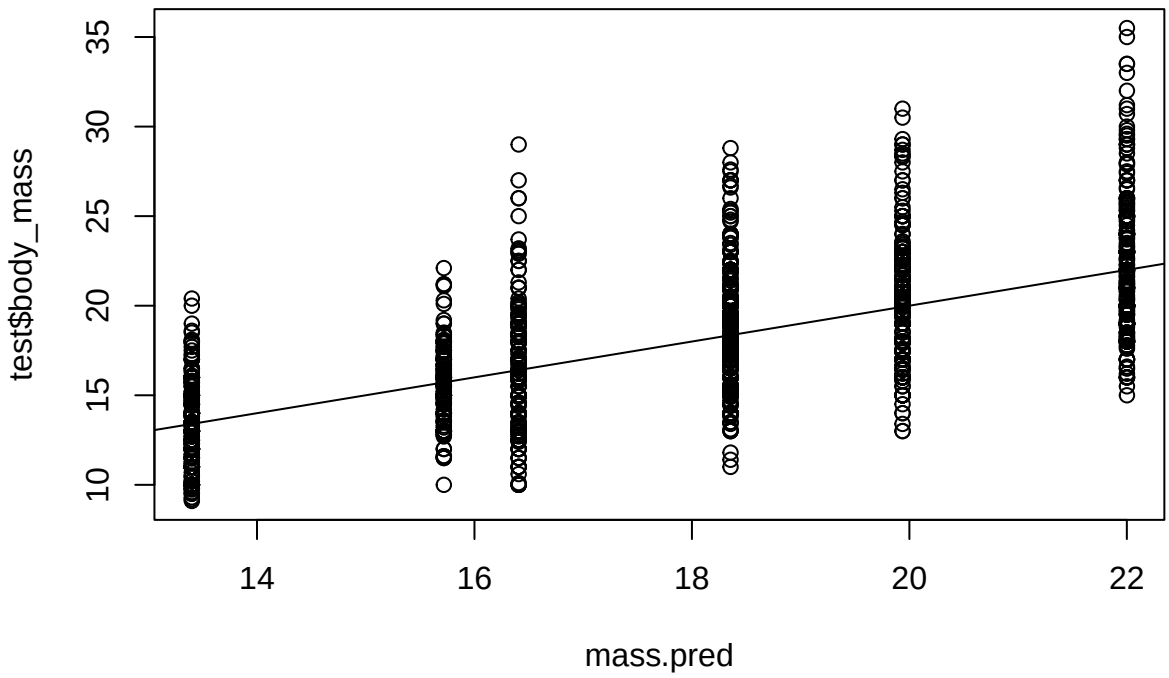
For individuals with a lifestage of “SUBAD” or “YOUNG”, the tree split on “total_length”, with individuals with a total length less than 149.5 having a predicted body mass of 13.40 and individuals with a total length

greater than 149.5 having a predicted body mass of 15.72. For individuals with a lifestage of "AD", the tree split on "HB.Length". Individuals with an HB.Length less than 81.75 had a predicted body mass of 16.41, while individuals with an HB.Length greater than 81.75 had a predicted body mass of 18.36. For individuals with an HB.Length greater than 88.5, the tree split on "total_length", with individuals with a total length less than 162.5 having a predicted body mass of 19.94 and individuals with a total length greater than 162.5 having a predicted body mass of 22.00.

Prediction and Evaluation:

After building the regression tree using the training data, the model was used to make predictions on the test data using the predict function.

```
mass.pred<-predict(tree.mass,test)
plot(mass.pred,test$body_mass)
abline(0,1)
```



The predicted values were plotted against the actual body mass values in a scatter plot to evaluate the performance of the tree. The plot shows a strong positive linear relationship between the predicted and actual body mass values, indicating that the model is performing well.

```
sqrt(mean((mass.pred-test$body_mass)^2))
```

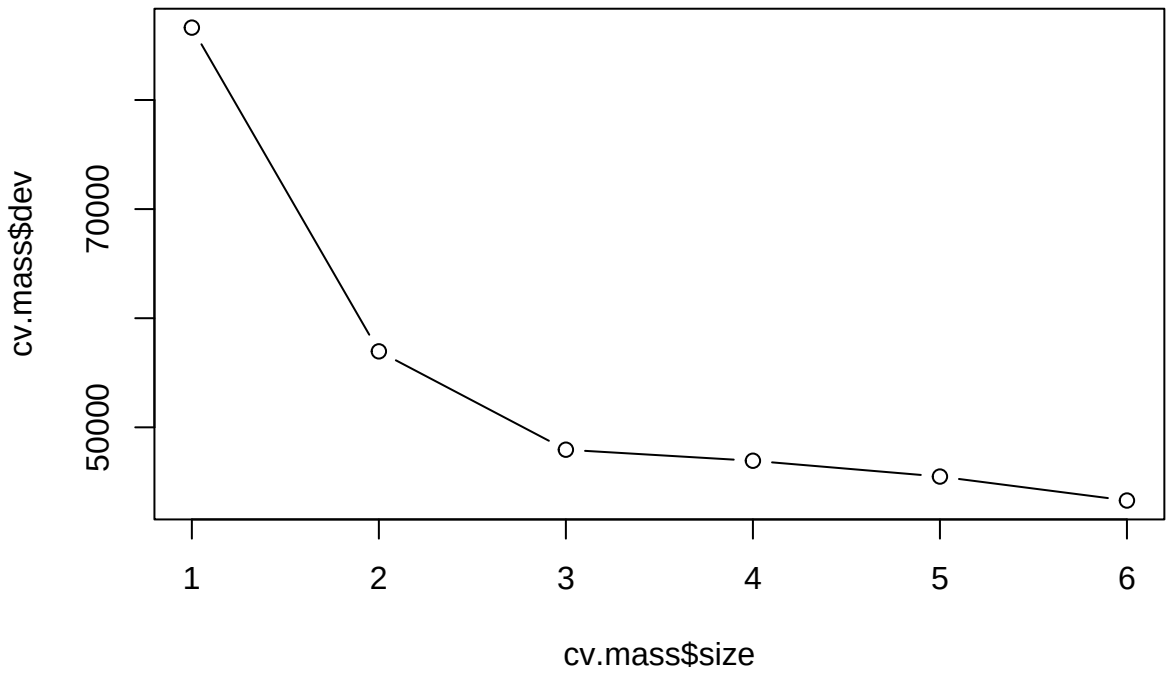
```
## [1] 3.024804
```

The performance of the tree was evaluated by making predictions on the test data and calculating RMSE between the predictions and the actual body mass values. The RMSE for the tree was 3.024804, which indicates that the tree is a reasonably good model for predicting body mass values.

Pruning

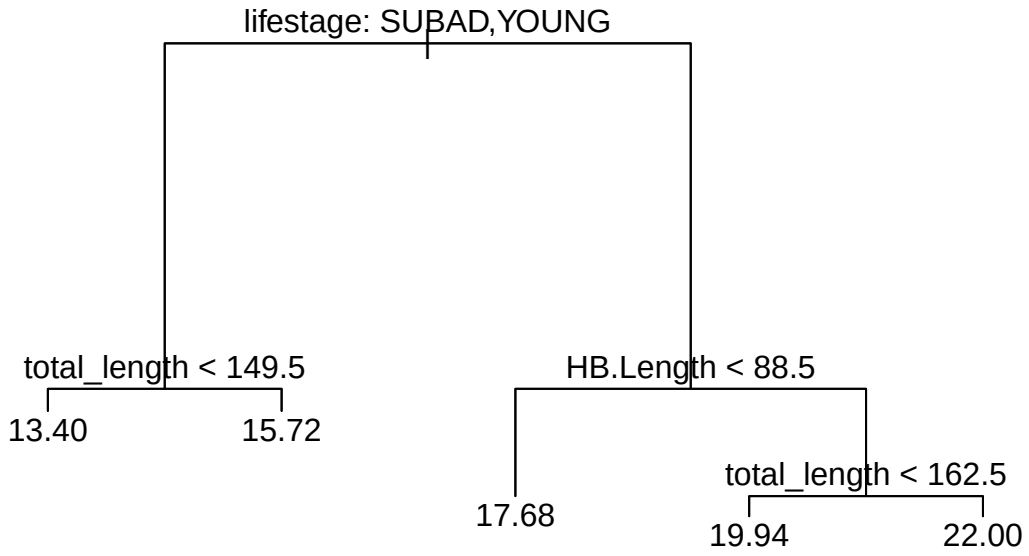
In order to reduce the complexity of the tree, the tree was pruned using the prune.tree function. The cross-validation error was plotted against the size of the tree to determine the "best" number of terminal nodes.

```
cv.mass<-cv.tree(tree.mass)
plot(cv.mass$size,cv.mass$dev,type="b")
```



Based on this plot, the best number of terminal nodes was found to be 5. This means that a tree with 5 terminal nodes had the best balance between overfitting and underfitting.

```
prune.mass=prune.tree(tree.mass,best=5)
plot(prune.mass)
text(prune.mass,pretty=0)
```

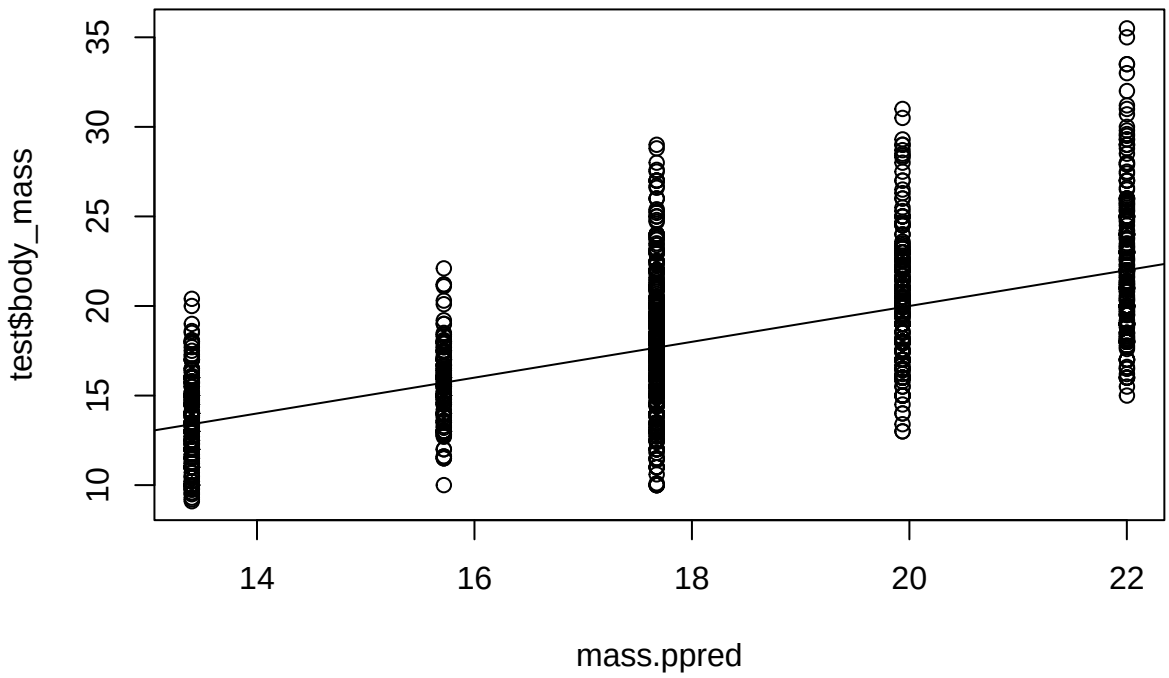


The pruned tree was built using three predictor variables: "lifestage", "total_length", and "HB.Length". The pruned tree has 5 terminal nodes, which indicates that the tree has 5 branches that stop at a final prediction. The first split of the tree is based on the "lifestage" variable, with two terminal nodes for "SUBAD, YOUNG" and "AD". The second split of the tree is based on the "total_length" variable for the "SUBAD, YOUNG" terminal node, with one additional terminal node for body mass values less than 149.5. The second split of the tree is based on the "total_length" variable for the "AD" terminal node, with two additional terminal nodes for body mass values greater than or less than 149.5. The third split of the tree is based on the "HB.Length" variable for the "AD" terminal node with body mass values less than or equal to 18.3.

```
summary(prune.mass)
```

```
##
## Regression tree:
## snip.tree(tree = tree.mass, nodes = 6L)
## Variables actually used in tree construction:
## [1] "lifestage"    "total_length" "HB.Length"
## Number of terminal nodes:  5
## Residual mean deviance:  8.726 = 44040 / 5047
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -10.7400  -1.9370  -0.2184   0.0000   1.5990  14.2600
```

```
mass.ppred<-predict(prune.mass,test)
plot(mass.ppred,test$body_mass)
abline(0,1)
```




```
sqrt(mean((mass.ppred-test$body_mass)^2)) #RMSE
```

```
## [1] 3.090389
```

The performance of the pruned model was evaluated by making predictions on the test data and calculating the root mean squared error (RMSE) between the predicted values and the actual body mass values. The RMSE for the pruned model was 3.090389.

4.5. Regression tree - total length (Author: Zheyu Song)

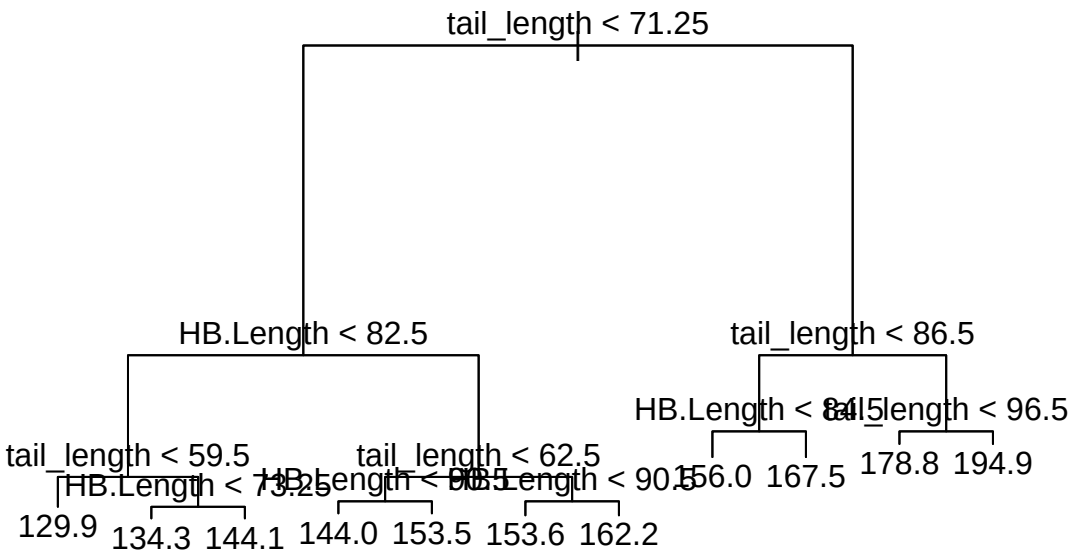
The process for building the regression tree model for body length is similar to the one described above for the body mass model. A regression tree was built using the body length as the target variable and the remaining variables as predictors.

```
tree.length<-tree(total_length ~. , data=train)
summary(tree.length)
```

```
##
## Regression tree:
## tree(formula = total_length ~ ., data = train)
## Variables actually used in tree construction:
## [1] "tail_length" "HB.Length"
## Number of terminal nodes:  11
```

```
## Residual mean deviance:  27.65 = 139400 / 5041
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -39.00000  -3.34500   0.03326   0.00000   3.37100  27.49000
```

```
plot(tree.length)
text(tree.length ,pretty =0)
```

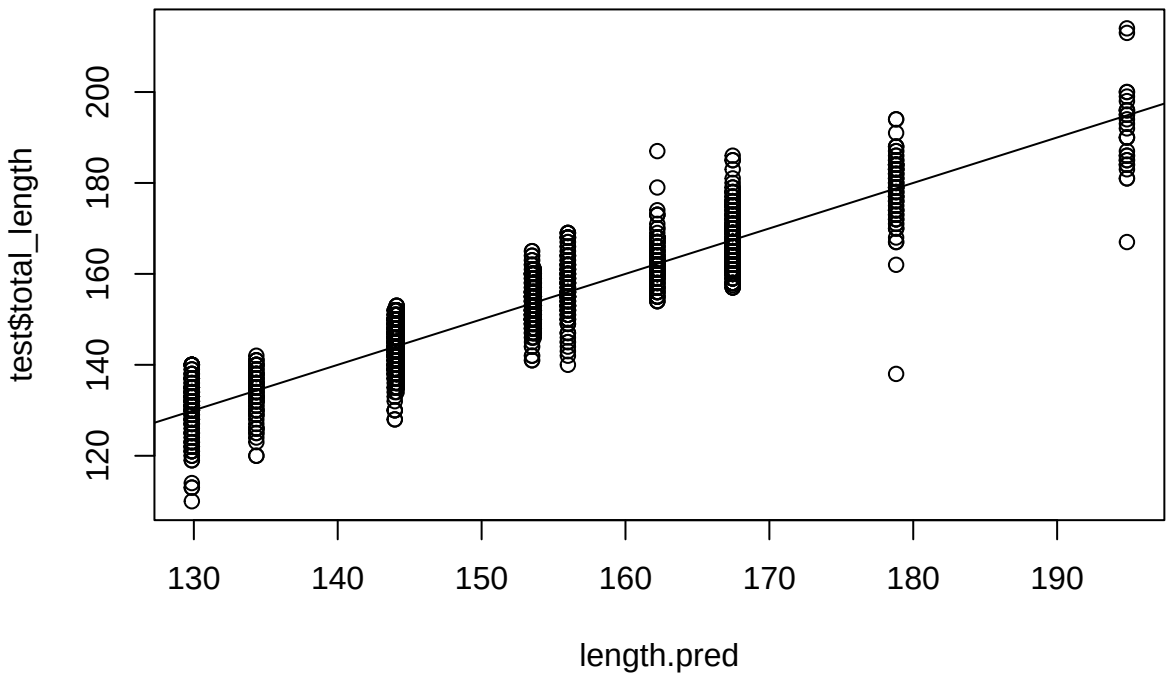


The tree consists of 11 terminal nodes and has a residual mean deviance of 27.65, which is the average of the squared differences between the predicted and actual values. The distribution of residuals ranges from -39 to 27.49, with a mean of 0. The first split in the tree is based on tail_length, with values less than 71.25 going to the left branch and those greater than or equal to 71.25 going to the right branch. The left branch further splits based on HB.Length, with values less than 82.5 going to a terminal node with a mean total length of 129.9 and values greater than or equal to 82.5 going to a node with a mean total length of 153.3. The right branch splits based on tail_length and HB.Length, resulting in three terminal nodes with mean total lengths of 144.0, 153.5, and 162.2, respectively. The interpretation of this tree suggests that tail_length is the most important predictor of total_length, followed by HB.Length.

Prediction and Evaluation:

The model was used to make predictions on the test data using the predict function.

```
length.pred<-predict(tree.length,test)
plot(length.pred,test$total_length)
abline(0,1)
```




```
sqrt(mean((length.pred-test$total_length)^2)) #RMSE
```

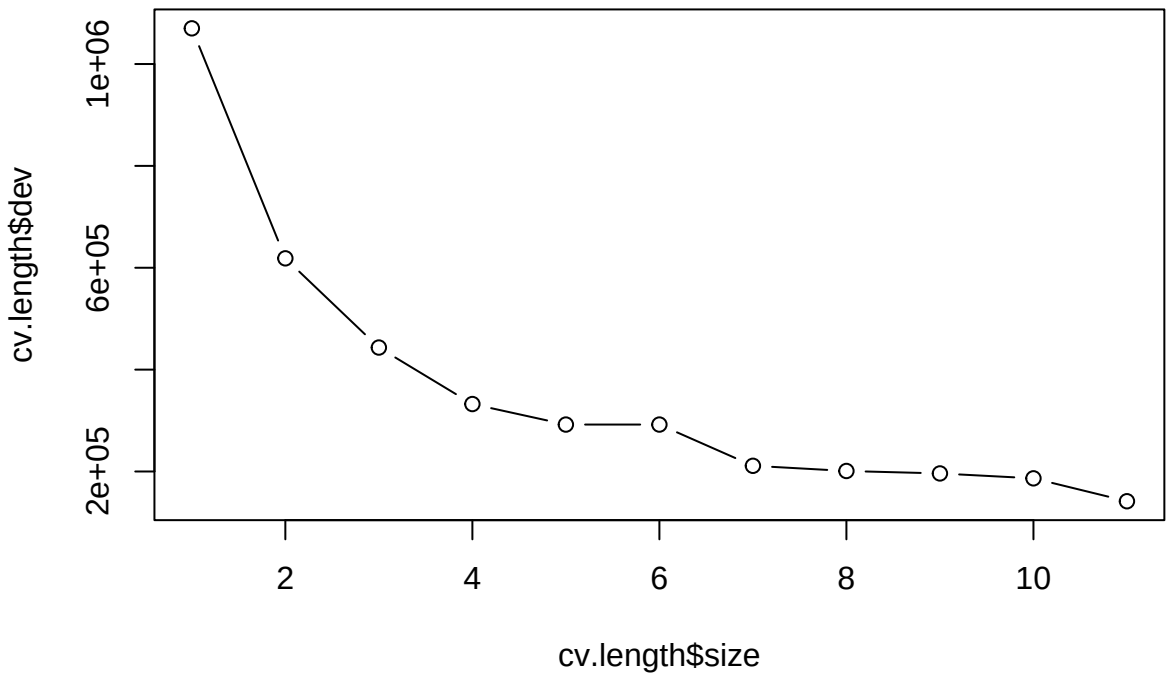
```
## [1] 5.308092
```

The performance of the model was evaluated using the root mean squared error (RMSE) metric, which measures the difference between the predicted and actual values of total length. The RMSE for the model on the test data was 5.308092, indicating that the model has a moderate level of accuracy in predicting total length.

Pruning

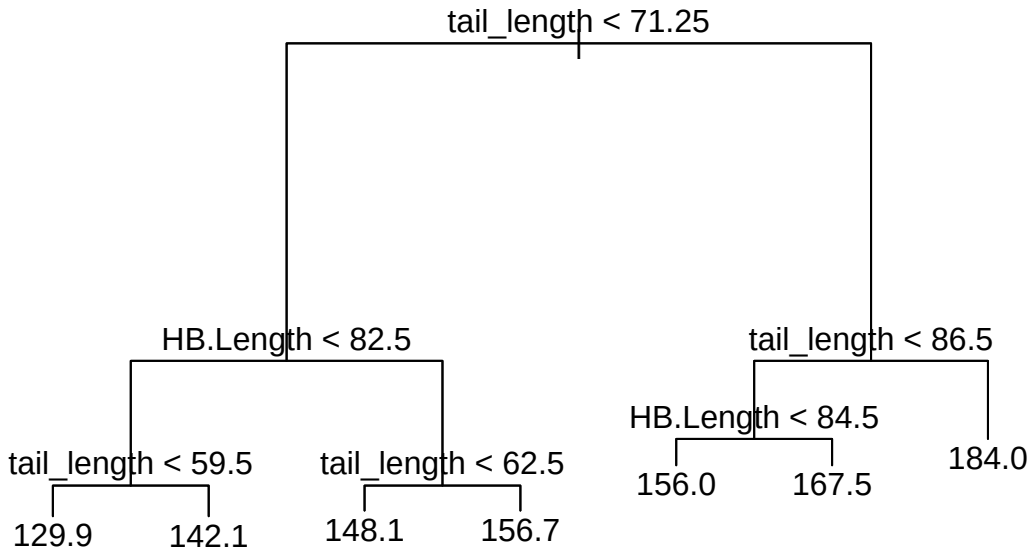
In order to reduce the complexity of the tree, the tree was pruned using the prune.tree function. The cross-validation error was plotted against the size of the tree to determine the "best" number of terminal nodes.

```
cv.length<-cv.tree(tree.length)
plot(cv.length$size,cv.length$dev,type="b")
```



Based on this plot, the best number of terminal nodes was found to be 7.

```
prune.length=prune.tree(tree.length,best=7)
plot(prune.length)
text(prune.length,pretty=0)
```




```
prune.length
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 5052 1070000 153.4
##    2) tail_length < 71.25 3494  388600 147.1
##      4) HB.Length < 82.5 1493   96730 138.9
##        8) tail_length < 59.5 395   15260 129.9 *
##        9) tail_length > 59.5 1098   37850 142.1 *
##      5) HB.Length > 82.5 2001  114400 153.3
##       10) tail_length < 62.5 789   42690 148.1 *
##       11) tail_length > 62.5 1212   36160 156.7 *
##    3) tail_length > 71.25 1558  229300 167.6
##      6) tail_length < 86.5 1233   82890 163.3
##       12) HB.Length < 84.5 451   18330 156.0 *
##       13) HB.Length > 84.5 782   27040 167.5 *
##      7) tail_length > 86.5 325   35380 184.0 *
```

```
summary(prune.length)
```

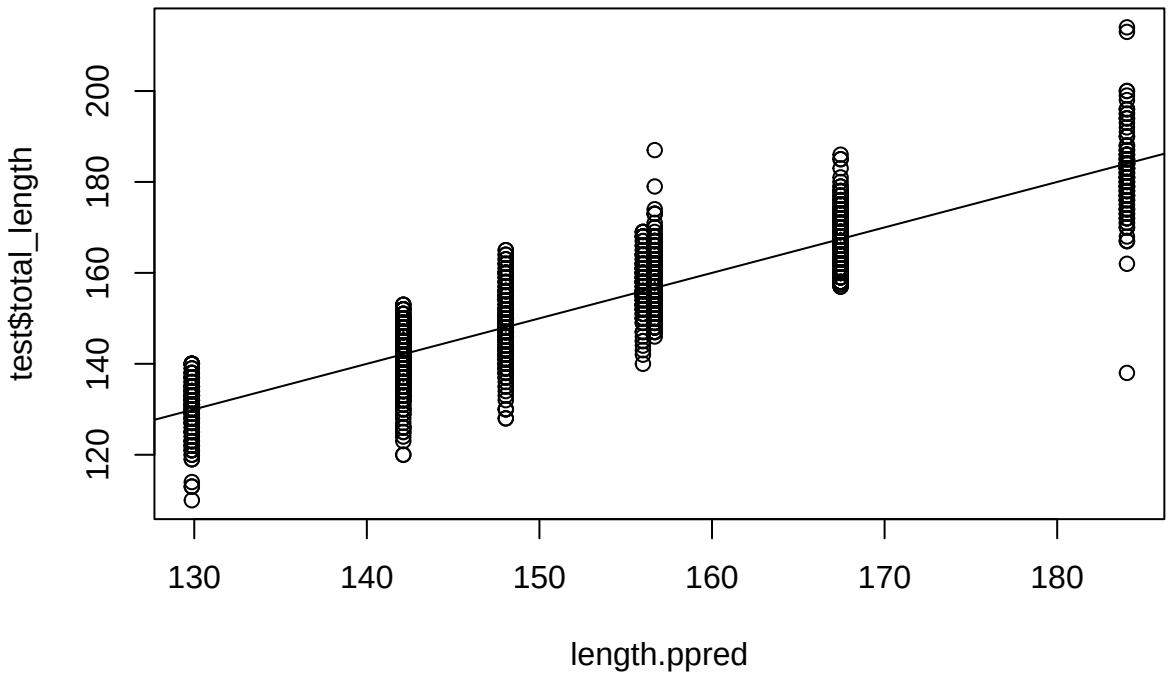
```
##
## Regression tree:
## snip.tree(tree = tree.length, nodes = c(9L, 10L, 7L, 11L))
## Variables actually used in tree construction:
```

```
## [1] "tail_length" "HB.Length"
## Number of terminal nodes:  7
## Residual mean deviance:  42.16 = 212700 / 5045
## Distribution of residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -39.000  -4.053  -0.114   0.000   3.947  32.950
```

The pruned regression tree for predicting total length based on tail length and HB.Length has a root node with 5052 observations. The first split is based on tail length, with observations with tail length less than 71.25 cm going to the left branch and those with tail length greater than or equal to 71.25 cm going to the right branch. In the left branch, the second split is based on HB.Length, with observations with HB.Length less than 82.5 cm going to the left subtree and those with HB.Length greater than or equal to 82.5 cm going to the right subtree. In the right branch, the second split is also based on HB.Length, with observations with HB.Length less than 84.5 cm going to the left subtree and those with HB.Length greater than or equal to 84.5 cm going to the right subtree.

The pruned tree has seven terminal nodes, with tail length and HB.Length being the only variables used in the tree construction. The residual mean deviance is 42.16, indicating that the model explains a significant portion of the variance in the total length data.

```
length.ppred<-predict(prune.length,test)
plot(length.ppred,test$total_length)
abline(0,1)
```



```
sqrt(mean((length.ppred-test$total_length)^2)) #RMSE
```

```
## [1] 6.457926
```

The pruned tree was then built and its performance was evaluated using the same metrics as the unpruned tree. The MSE for the pruned tree was found to be 6.457926.

4.6. Regression Tree Models for Predicting Total Length without Length Variables (Author: Zheyu Song)

We have decided to drop the "tail_length" and "HB.Length" variables from our dataset for predicting the total length of the deer mouse. Our decision is based on the fact that the tree constructed using these variables has already shown a strong relationship with the target variable. We believe that it is unnecessary to include other length measurements of the body for predicting the total length if the "tail_length" and "HB.Length" variables can be measured directly. Therefore, we have rebuilt a regression tree model to predict the total length using a dataset without these two variables.

```
df1 <- subset(df, select = -c(tail_length, HB.Length))
set.seed (10)
splitIndex <- sample(1:nrow(df1), size = 3/4 * nrow(df1))
train1 <- df1[splitIndex, ]
test1 <- df1[-splitIndex, ]
#head(train)
dim(test1)
```

```
## [1] 1685   18
```

```
dim(train1)
```

```
## [1] 5052   18
```

```
names(df1)
```

```
##  [1] "pop_density_4km2" "season"           "lifestage"        "sp"
##  [5] "sex"              "body_mass"        "total_length"     "MAT"
##  [9] "MWMT"             "MCMT"             "TD"               "MAP"
## [13] "MSP"              "DD5"              "FFP"              "EMT"
## [17] "EXT"              "ecoregion1"
```

```
tree.length1<-tree(total_length ~. , data=train1)

summary(tree.length1)
```

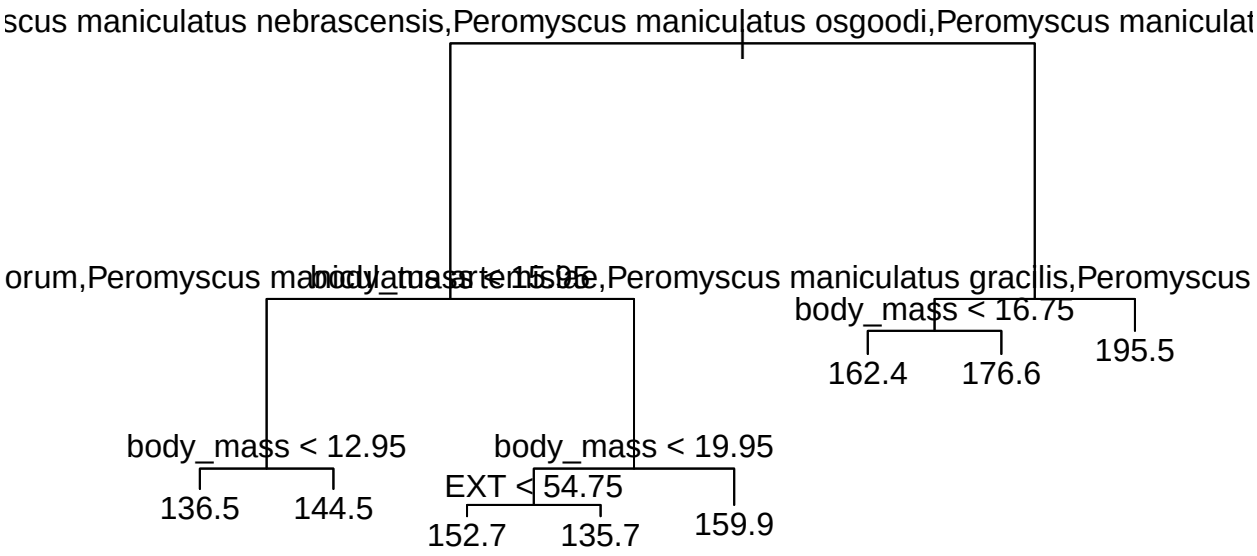
```
##
## Regression tree:
## tree(formula = total_length ~ ., data = train1)
## Variables actually used in tree construction:
## [1] "sp"        "body_mass" "EXT"
## Number of terminal nodes:  8
```

```
## Residual mean deviance:  88.5 = 446400 / 5044
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -44.3600  -5.6670   0.1347   0.0000   6.1350  37.3300
```

```
tree.length1
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 5052 1070000 153.4
##    2) sp: Peromyscus maniculatus bairdii,Peromyscus maniculatus blandus,Peromyscus maniculatus gambel
##      4) body_mass < 15.95 1597  150900 141.8
##        8) body_mass < 12.95 540   42390 136.5 *
##        9) body_mass > 12.95 1057   85280 144.5 *
##      5) body_mass > 15.95 2838  307500 155.6
##       10) body_mass < 19.95 1591  143300 152.2
##         20) EXT < 54.75 1545  124600 152.7 *
##         21) EXT > 54.75 46    5891 135.7 *
##       11) body_mass > 19.95 1247  122900 159.9 *
##    3) sp: Peromyscus maniculatus abietorum,Peromyscus maniculatus artemisiae,Peromyscus maniculatus a
##      6) sp: Peromyscus maniculatus abietorum,Peromyscus maniculatus artemisiae,Peromyscus maniculatus
##       12) body_mass < 16.75 213   22690 162.4 *
##       13) body_mass > 16.75 336   37590 176.6 *
##      7) sp: Peromyscus maniculatus austerus,Peromyscus maniculatus oreas 68    5047 195.5 *
```

```
plot(tree.length1)
text(tree.length1 ,pretty =0)
```

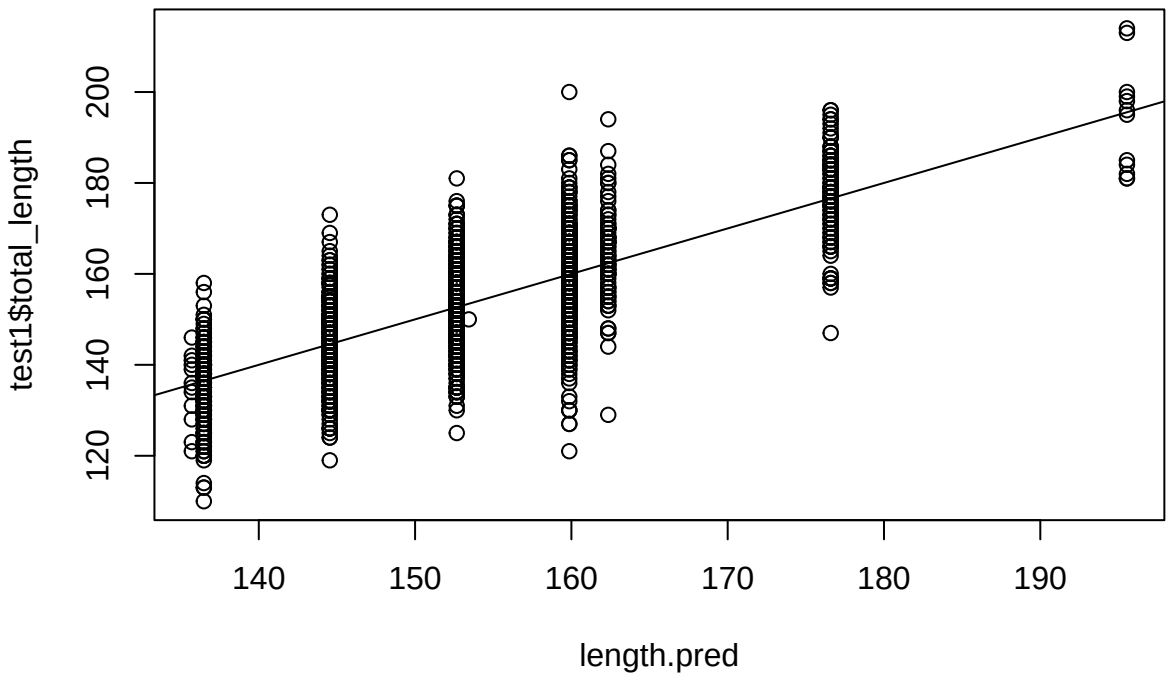


We rebuilt the regression tree model for predicting total length using the deer mouse dataset, train1, without the length variables.

The regression tree model for predicting total length using the deer mouse dataset without "tail_length" and "HB.Length" variables produced a tree with 8 terminal nodes. The variables used in tree construction were "sp," "body_mass," and "EXT."

The regression tree showed that the "body_mass" variable played a critical role in determining the total length of the deer mouse. Additionally, the "EXT" variable, which measures the ear-to-body length ratio, was used in some of the splits. The tree indicates that the deer mouse species "Peromyscus maniculatus abietorum," "Peromyscus maniculatus artemisiae," "Peromyscus maniculatus gracilis," "Peromyscus maniculatus hollisteri," "Peromyscus maniculatus nubiterrae," and "Peromyscus maniculatus rubidus" have a higher mean total length compared to the other species in the dataset.

```
length.pred<-predict(tree.length1,test1)
plot(length.pred,test1$total_length)

abline(0,1)
```



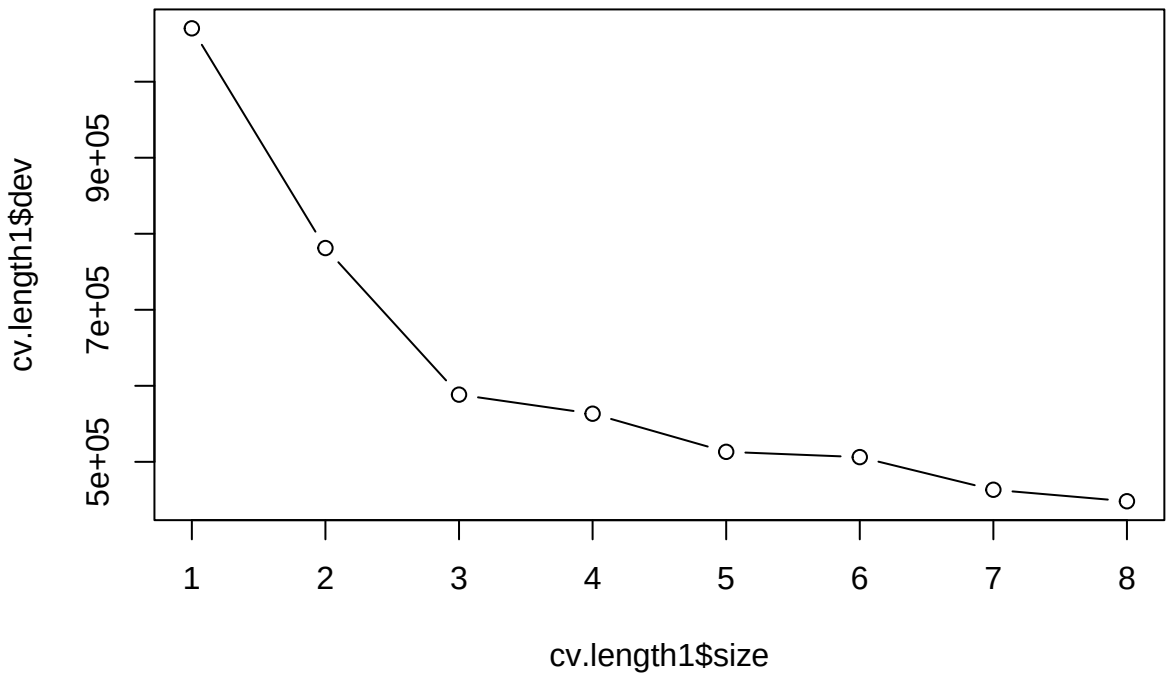

```
sqrt(mean((length.pred-test1$total_length)^2)) #the mean squared error
```

```
## [1] 9.284538
```

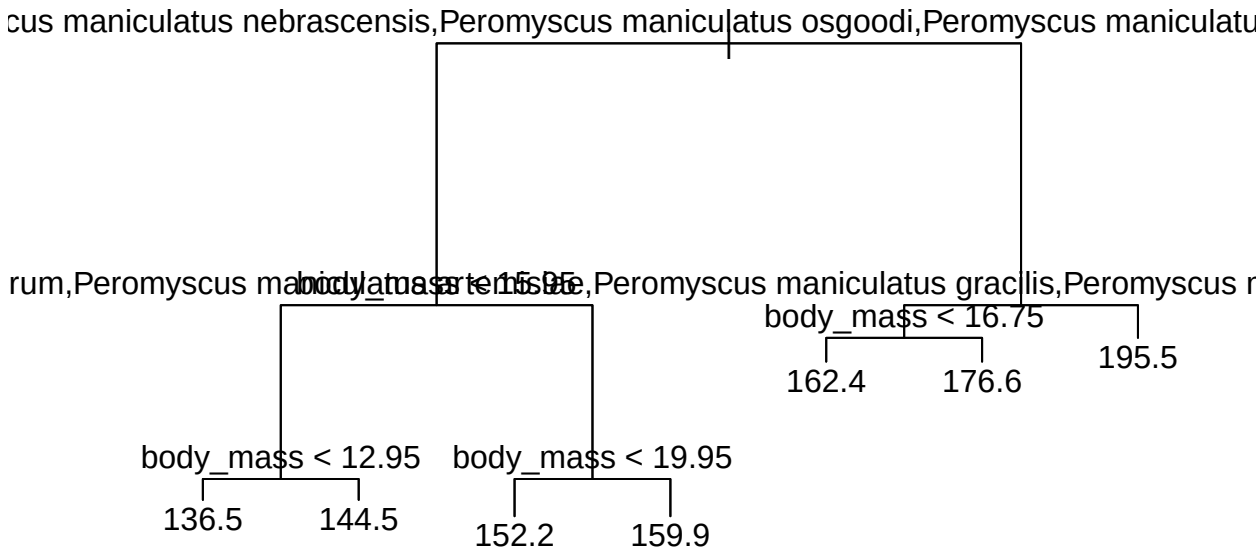
The tree's performance was evaluated using the same metrics as the previous model, and the root MSE was found to be 9.284538

Pruning

```
cv.length1<-cv.tree(tree.length1)
plot(cv.length1$size,cv.length1$dev,type="b")
```



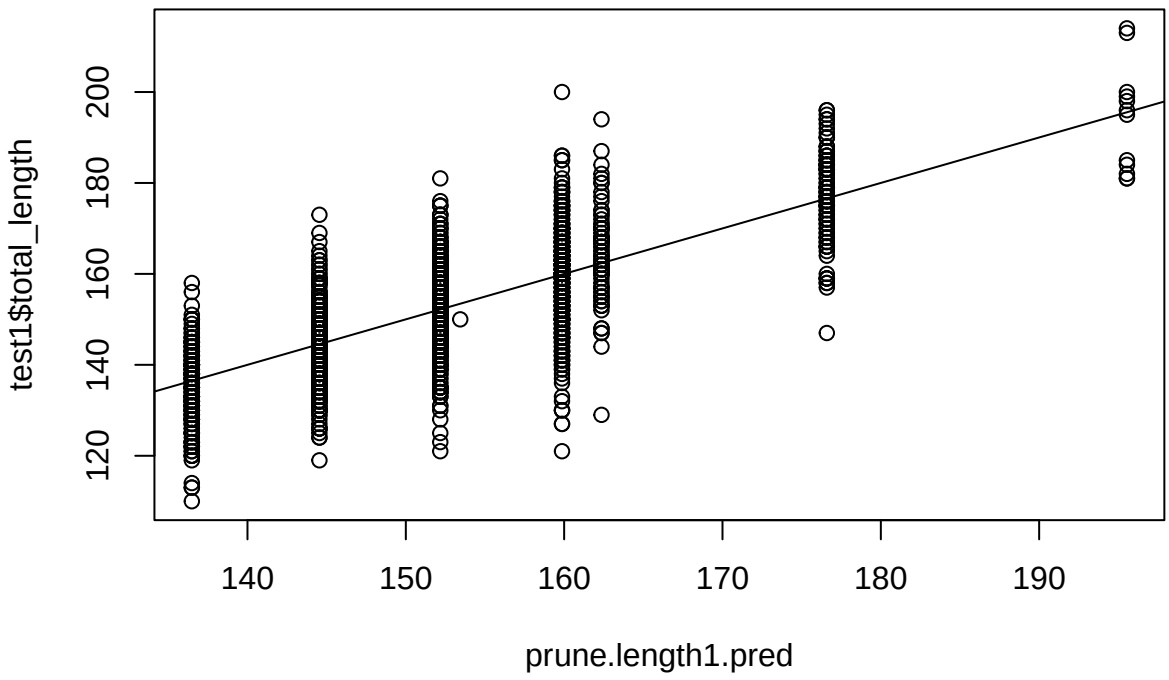

```
prune.length1=prune.tree(tree.length1,best=7)
plot(prune.length1)
text(prune.length1,pretty=0)
```




```
prune.length1
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 5052 1070000 153.4
##    2) sp: Peromyscus maniculatus bairdii,Peromyscus maniculatus blandus,Peromyscus maniculatus gambel
##      4) body_mass < 15.95 1597  150900 141.8
##        8) body_mass < 12.95 540   42390 136.5 *
##        9) body_mass > 12.95 1057   85280 144.5 *
##      5) body_mass > 15.95 2838  307500 155.6
##       10) body_mass < 19.95 1591  143300 152.2 *
##       11) body_mass > 19.95 1247  122900 159.9 *
##    3) sp: Peromyscus maniculatus abietorum,Peromyscus maniculatus artemisiae,Peromyscus maniculatus a
##      6) sp: Peromyscus maniculatus abietorum,Peromyscus maniculatus artemisiae,Peromyscus maniculatus
##       12) body_mass < 16.75 213   22690 162.4 *
##       13) body_mass > 16.75 336   37590 176.6 *
##      7) sp: Peromyscus maniculatus austerus,Peromyscus maniculatus oreas 68    5047 195.5 *
```

```
prune.length1.pred<-predict(prune.length1,test1)
plot(prune.length1.pred,test1$total_length)

abline(0,1)
```




```
sqrt(mean((prune.length1.pred-test1$total_length)^2)) #RMSE
```

```
## [1] 9.409149
```

We then pruned the tree and found that the pruned tree has a slightly higher RMSE of 9.409149. This is higher than the RMSE for the model that includes the length variables, indicating that the length variables improve the model's ability to predict total length. This may be due to the fact that body mass and hind foot length are correlated with total length. We concluded that the attempt to remove the length variables was not successful, as the resulting model had a much higher RMSE compared to the previous model. Therefore, we decided to continue using the previous model that included all variables, as it was more effective in predicting the total length of deer mice.

4.7. Cross-Validation Results for Regression Tree Models on Body Mass and Total Length (Author: Zheyu Song)

In regression analysis, it is important to assess the accuracy of a model's predictions before applying it to new data. In this analysis, we used k-fold cross-validation to evaluate the performance of regression tree models on predicting the body mass and total length of deer mice.

For each model, we split the data into training and validation sets for each fold, trained the pruned regression tree model on the training data, and evaluated the model on the validation data. We then calculated the root mean square error (RMSE) for each fold and averaged the RMSE values across all folds to obtain an estimate of the model's performance.

Regression Tree Model for Body Mass

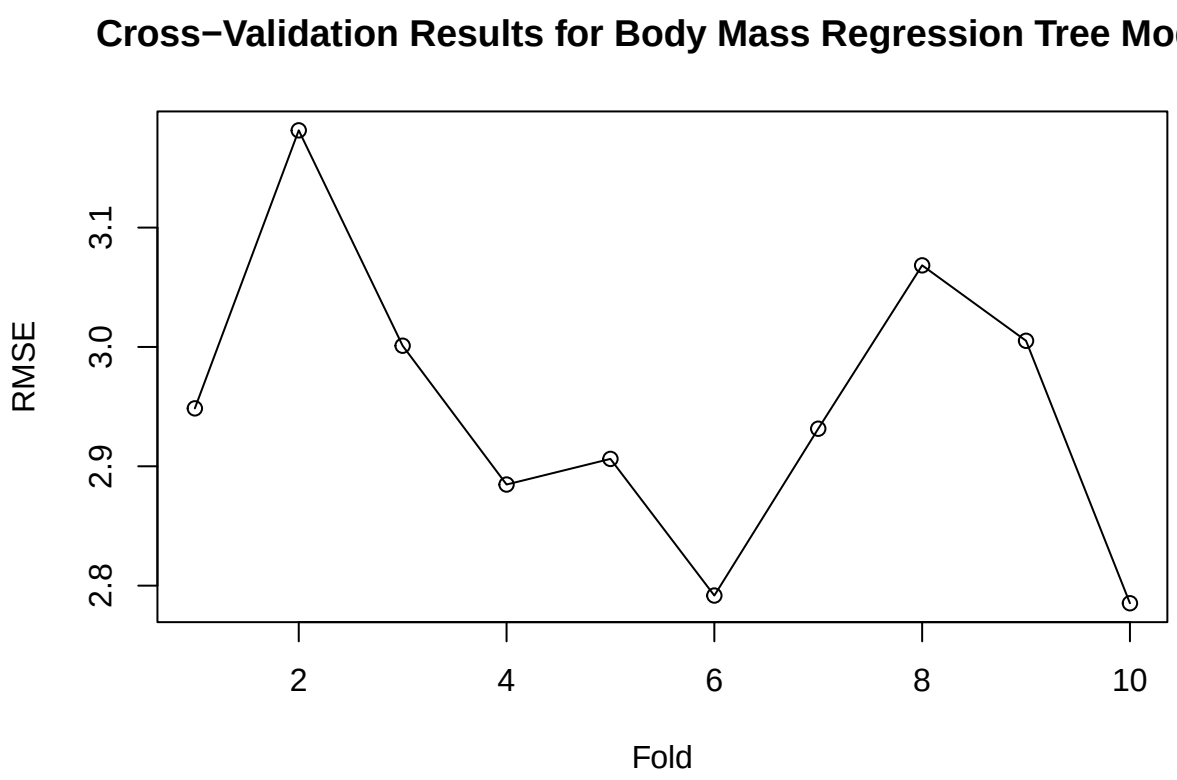
```
library(caret)
set.seed (10)

# Specify the number of folds for cross-validation
k <- 10

# Create a k-fold cross-validation object
folds <- createFolds(train$body_mass, k = k)

# Initialize a vector to store the performance metrics for each fold
performance_metrics <- rep(0, k)

# Loop through each fold, training and evaluating the model on each one
for (i in 1:k) {
  # Split the training data into training and validation sets for this fold
  train_indices <- unlist(folds[-i])
  valid_indices <- folds[[i]]
  train_data <- train[train_indices, ]
  valid_data <- train[valid_indices, ]

  # Train the pruned regressor model on the training data
  model <- prune.mass

  # Evaluate the model on the validation data
  predictions <- predict(model, newdata = valid_data)
  performance_metric <- sqrt(mean((predictions - valid_data$body_mass)^2))

  # Store the performance metric for this fold
  performance_metrics[i] <- performance_metric
}

# Compute the average performance metric across all folds
average_performance <- mean(performance_metrics)
performance_metrics
```

```
##  [1] 2.948533 3.181484 3.001022 2.884735 2.906186 2.791717 2.931497 3.068355
##  [9] 3.005158 2.785257
```

```
average_performance
```

```
## [1] 2.950394
```

```
plot(performance_metrics, type = "o", xlab = "Fold", ylab = "RMSE", main = "Cross-Validation Results fo
```



The resulting average performance metric was 2.950394, which indicates good predictive accuracy of the model for body mass of deer mice. The performance metrics for each fold ranged from 2.785257 to 3.181484, which also suggests that the model is consistent in its predictions across different folds.

Regression Tree Model for Predicting Total Length (Author: Zheyu Song)

```
k <- 10
set.seed (10)

folds <- createFolds(train$total_length, k = k)

performance_metrics <- rep(0, k)

for (i in 1:k) {
  train_indices <- unlist(folds[-i])
  valid_indices <- folds[[i]]
  train_data <- train[train_indices, ]
  valid_data <- train[valid_indices, ]

  model <- prune.length

  predictions <- predict(model, newdata = valid_data)
  performance_metric <- sqrt(mean((predictions - valid_data$total_length)^2))

  performance_metrics[i] <- performance_metric
```

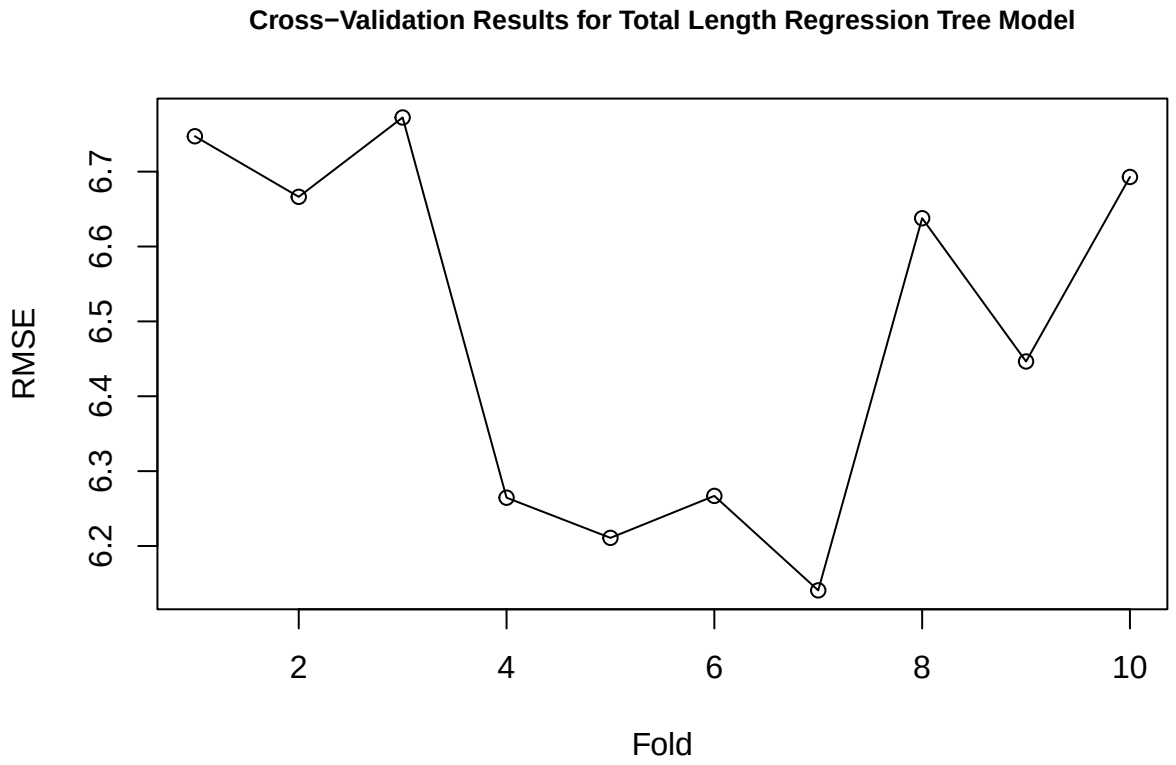
```
}

average_performance <- mean(performance_metrics)
performance_metrics
```

```
##  [1] 6.747272 6.666371 6.772342 6.264517 6.210873 6.266901 6.140738 6.637783
##  [9] 6.446455 6.692891
```

```
average_performance
```

```
## [1] 6.484614
```

```
plot(performance_metrics, type = "o", xlab = "Fold", ylab = "RMSE", main = "Cross-Validation Results fo
```



The results for the regression tree model on total length of deer mice show that the average performance metric across all folds was 6.48. This indicates that the model has a relatively high error in predicting the total length of deer mice, as the predicted values are, on average, around 6.48 units away from the true values.

The performance metric for each fold ranges from 6.14 to 6.77, which suggests that the model's accuracy may vary depending on the subset of the data used for training and testing. Overall, it appears that the regression tree model for total length may not be as effective as the model for body mass, as the average performance metric is higher for total length than it is for body mass.

Regression Tree Models for Predicting Total Length without Length Variables

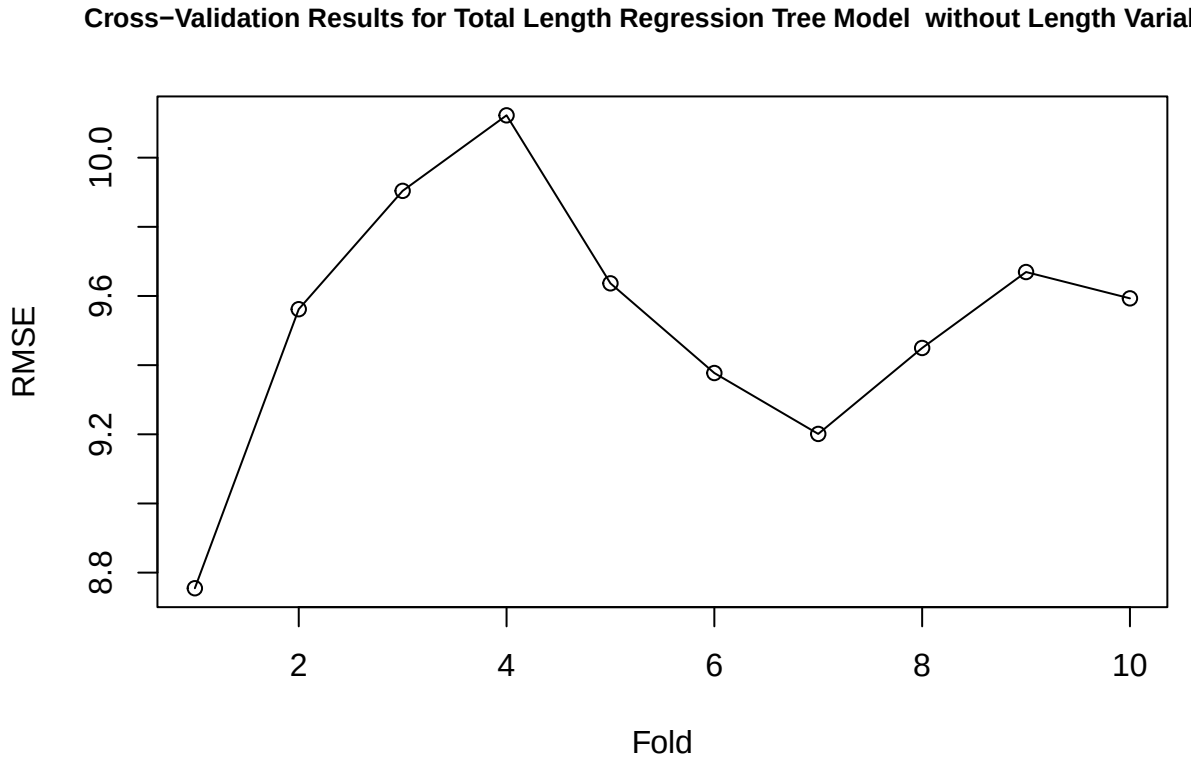
```
k <- 10
set.seed (10)

folds <- createFolds(train1$total_length, k = k)

performance_metrics <- rep(0, k)

for (i in 1:k) {
  train_indices <- unlist(folds[-i])
  valid_indices <- folds[[i]]
  train_data <- train[train_indices, ]
  valid_data <- train[valid_indices, ]

  model <- prune.length1

  predictions <- predict(model, newdata = valid_data)
  performance_metric <- sqrt(mean((predictions - valid_data$total_length)^2))

  performance_metrics[i] <- performance_metric
}

average_performance <- mean(performance_metrics)
performance_metrics
```

```
##  [1]  8.754835  9.561758  9.903990 10.122383  9.636549  9.377207  9.201090
##  [8]  9.449667  9.669278  9.593170
```

```
average_performance
```

```
## [1] 9.526993
```

```
plot(performance_metrics, type = "o", xlab = "Fold", ylab = "RMSE", main = "Cross-Validation Results fo
```



For the regression tree model for predicting the total length without length variables, we see that the average RMSE across all folds is 9.53. This is higher than the average RMSE for the model that includes the length variables, indicating that the length variables improve the model's ability to predict total length. This may be due to the fact that body mass and hind foot length are correlated with total length, and by including these variables, we provide the model with more information about the deer mice that may improve its accuracy.

4.8. Comparison of regression methods for predicting body mass and total length. (Author: Maciej Pecak)

- Body mass - CV error:
  - Linear regression - 2.5681 - (assumptions not met)
  - Regression tree - 2.9504
- Total length - CV error:
  - Linear regression - 6.2204 - (assumptions not met)
  - Regression tree - 6.48 - (including tail length / HB.Length)
  - Regression tree - 9.53 - (without tail length / HB.Length)

As presented above, in both cases cross validation error for linear regression model is smaller, however both models don't meet the required assumptions expected for residuals (normality/homoscedasticity) which means using linear regression for further predictions might not yield correct results. In this case, it's probably better to use regression tree models, because they don't need to follow any assumptions.

5. Climate impact on mice body mass and total length. (Authors: Maciej Pecak, Zheyu Song)

This part of the study aimed to investigate the relationship between climate and body size in mammals. Understanding the drivers of body size variation in mammals is crucial for predicting how they will respond to climate change, and for understanding the ecological and evolutionary processes that have shaped their diversity.

The study focused on five climate variables: temperature seasonality (TD), mean annual precipitation (MAP), mean summer precipitation (MSP), frost-free period (FFP), and extreme temperature (EXT). These variables were selected because they are known to be important drivers of ecological processes and have been shown to affect body size in previous studies.

We investigated the impact of climate on body size. We selected the relevant climate variables (TD, MAP, MSP, FFP, EXT) for both the body mass and total length data frames and used linear regression and regression trees to build models.

5.1. Linear regression

We read the data first to have the data frame fresh.

```
mice.df <- read.csv("mice_filled_all_values.csv") %>%
  mutate(
    season = as.factor(season),
    lifestage = as.factor(lifestage),
    sex = as.factor(sex),
    ecoregion1 = as.factor(ecoregion1)
  )
```

In order to keep the analysis consistent, only mice subspecies that were observed more than 100 times were taken into consideration. That also helps to reduce dimensionality.

```
species.considered <- c(
  "Peromyscus maniculatus Wagner, 1845",
  "Peromyscus maniculatus sonoriensis",
  "Peromyscus maniculatus bairdii",
  "Peromyscus maniculatus gambelii",
  "Peromyscus maniculatus artemisiae",
  "Peromyscus maniculatus rufinus",
  "Peromyscus maniculatus rubidus",
  "Peromyscus maniculatus nebrascensis",
  "Peromyscus maniculatus luteus"
)

model.df <- mice.df %>%
  select(-c(X.1, X, long, lat, decade, month, year, ecoregion1_num,
            season_num, sex_num, sex_transformed,
            ecoregion1_transformed, season_transformed)) %>%
  filter(sp %in% species.considered)
```

In order to ensure the correctness of the algorithm, the variables that are highly multicolinear need to be eliminated. Is was conducted manually by eliminating the correlated variables one by one, based on the highest value of the Variance Inflation Factor. The following subset contains values that are not colinear.

```
climate.bm.df <- model.df %>%
  filter(lifestage == "AD") %>%
  dplyr::select(c(body_mass, MAT, MWMT, MCMT, TD, MAP, MSP, DD5, FFP, EMT, EXT))

vif(lm(body_mass ~ ., data = climate.bm.df))
```

```
##          MAT         MWMT         MCMT           TD          MAP          MSP
##    70.336576  6218.167932 10230.738119  9519.152783     2.041199     3.485363
##          DD5          FFP          EMT          EXT
##    61.607302    12.244946    10.539607     8.142747
```

After eliminating the multicolinear variables:

```
climate.bm.df <- model.df %>%
  filter(lifestage == "AD") %>%
  dplyr::select(c(body_mass, TD, MAP, MSP, FFP, EXT))

vif(lm(body_mass ~ ., data = climate.bm.df))
```

```
##       TD      MAP      MSP      FFP      EXT
## 2.476552 1.647587 2.637010 2.614721 3.724225
```

Similar operation has been conducted with consideration of the total_length of the mouse.

```
climate.len.df <- model.df %>%
  filter(lifestage == "AD") %>%
  dplyr::select(c(total_length, MAT, MWMT, MCMT, TD, MAP, MSP, DD5, FFP, EMT, EXT))

vif(lm(total_length ~ ., data = climate.len.df))
```

```
##          MAT         MWMT         MCMT           TD          MAP          MSP
##    70.336576  6218.167932 10230.738119  9519.152783     2.041199     3.485363
##          DD5          FFP          EMT          EXT
##    61.607302    12.244946    10.539607     8.142747
```

```
climate.len.df <- model.df %>%
  filter(lifestage == "AD") %>%
  dplyr::select(c(total_length, TD, MAP, MSP, FFP, EXT))

vif(lm(total_length ~ ., data = climate.len.df))
```

```
##       TD      MAP      MSP      FFP      EXT
## 2.476552 1.647587 2.637010 2.614721 3.724225
```

Finally, the linear model for the total length based on the climate variables.

```
m.len <- lm(total_length ~ ., data = climate.len.df)
summary(m.len)
```

```
##
## Call:
## lm(formula = total_length ~ ., data = climate.len.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.079  -6.839  -0.041   7.303  45.262
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.742e+02  1.721e+00 101.186  < 2e-16 ***
## TD           2.648e-01  5.209e-02   5.085 3.85e-07 ***
## MAP          5.048e-03  6.040e-04   8.357  < 2e-16 ***
## MSP         -7.218e-04  1.816e-03  -0.398    0.691
## FFP          3.808e-02  4.351e-03   8.752  < 2e-16 ***
## EXT         -8.302e-01  5.493e-02 -15.116  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.14 on 4148 degrees of freedom
## Multiple R-squared:  0.1133, Adjusted R-squared:  0.1122
## F-statistic:   106 on 5 and 4148 DF,  p-value: < 2.2e-16
```

To increase the precision, the interaction terms have been added.

```
m.len <- lm(total_length ~ (TD + MAP + FFP + EXT) ^ 2, data = climate.len.df)
summary(m.len)
```

```
##
## Call:
## lm(formula = total_length ~ (TD + MAP + FFP + EXT)^2, data = climate.len.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -57.204  -6.698  -0.121   7.268  44.520
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.536e+02  9.011e+00  17.041  < 2e-16 ***
## TD           6.222e-01  3.218e-01   1.934 0.053215 .
## MAP          1.796e-02  4.032e-03   4.455 8.60e-06 ***
## FFP         -1.969e-02  3.212e-02  -0.613 0.539788
## EXT         -3.541e-02  2.522e-01  -0.140 0.888340
## TD:MAP      -2.364e-05  1.443e-04  -0.164 0.869830
## TD:FFP       9.851e-04  6.328e-04   1.557 0.119582
## TD:EXT      -1.385e-02  7.740e-03  -1.789 0.073661 .
## MAP:FFP      6.311e-05  1.319e-05   4.784 1.78e-06 ***
## MAP:EXT     -5.812e-04  1.622e-04  -3.583 0.000344 ***
## FFP:EXT     -6.516e-05  6.556e-04  -0.099 0.920832
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.08 on 4143 degrees of freedom
```

```
## Multiple R-squared:  0.1246, Adjusted R-squared:  0.1225
## F-statistic: 58.95 on 10 and 4143 DF,  p-value: < 2.2e-16
```

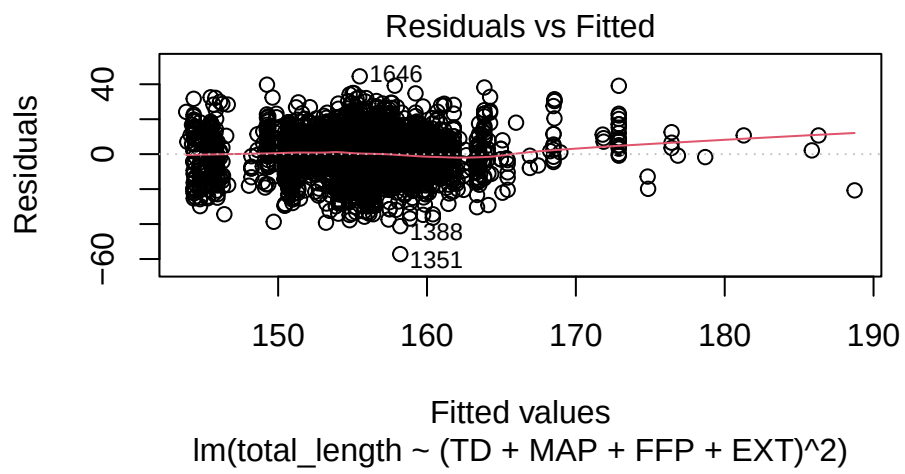
The computed RSME:

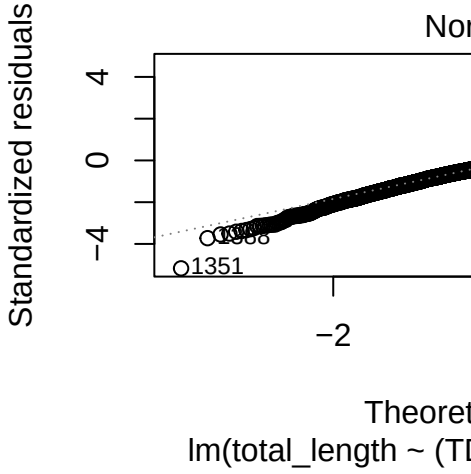
```
sqrt(mean(m.len$residuals ^ 2))
```

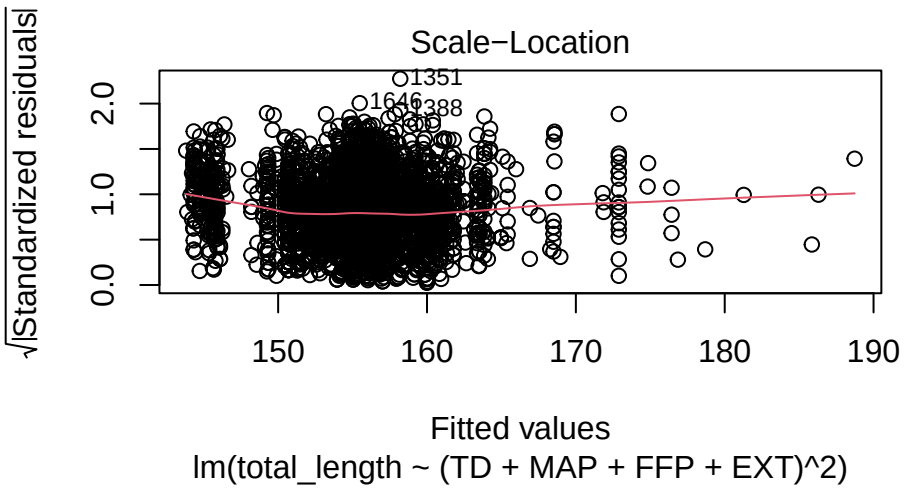
```
## [1] 11.06541
```

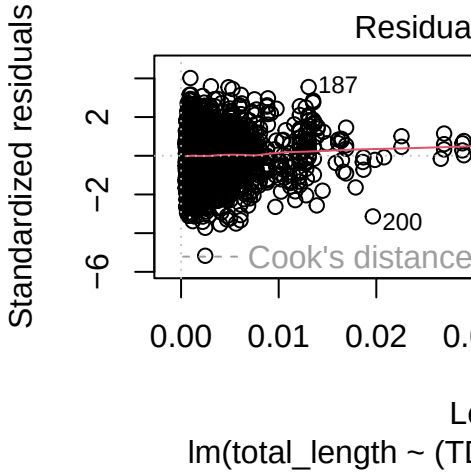
The visualization for the assumption inspection:

```
plot(m.len)
```









As we can see above by the visual inspection, the normality of residuals and the homoscedasticity of them looks correct. There are no significant outliers. measured by Cook's distance.

Similar approach was taken to predict the body mass.

```
m.bm <- lm(body_mass ~ ., data = climate.bm.df)
summary(m.bm)
```

```
##
## Call:
## lm(formula = body_mass ~ ., data = climate.bm.df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.2938  -2.5028  -0.3401   2.0603  16.2725
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 22.2205522  0.5715705  38.876  < 2e-16 ***
## TD          -0.0160426  0.0172943  -0.928  0.35366
## MAP         -0.0005286  0.0002005  -2.636  0.00843 **
## MSP          0.0053479  0.0006028   8.871  < 2e-16 ***
## FFP          0.0059217  0.0014448   4.099 4.23e-05 ***
## EXT         -0.1046261  0.0182371  -5.737 1.03e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.7 on 4148 degrees of freedom
## Multiple R-squared:  0.0197, Adjusted R-squared:  0.01852
## F-statistic: 16.67 on 5 and 4148 DF,  p-value: 2.424e-16
```

```
m.bm <- lm(body_mass ~ (MAP + MSP + FFP + EXT) ^ 2, data = climate.bm.df)
summary(m.bm)
```

```
##
## Call:
## lm(formula = body_mass ~ (MAP + MSP + FFP + EXT)^2, data = climate.bm.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.4298 -2.4454 -0.3122  2.0232 16.8018
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.667e+01  1.441e+00  25.447  < 2e-16 ***
## MAP         -6.442e-03  1.492e-03  -4.317 1.62e-05 ***
## MSP          9.040e-03  3.181e-03   2.842  0.00450 **
## FFP         -7.466e-02  9.844e-03  -7.584 4.11e-14 ***
## EXT         -4.801e-01  3.868e-02 -12.411  < 2e-16 ***
## MAP:MSP     -3.101e-06  1.064e-06  -2.914  0.00358 **
## MAP:FFP      9.506e-06  3.475e-06   2.735  0.00626 **
## MAP:EXT      1.313e-04  4.757e-05   2.761  0.00579 **
## MSP:FFP     -1.826e-06  1.455e-05  -0.126  0.90010
```

```
## MSP:EXT     -5.102e-05  9.759e-05  -0.523  0.60111
## FFP:EXT      1.939e-03  2.164e-04   8.959  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.646 on 4143 degrees of freedom
## Multiple R-squared:  0.04968,    Adjusted R-squared:  0.04739
## F-statistic: 21.66 on 10 and 4143 DF,  p-value: < 2.2e-16
```

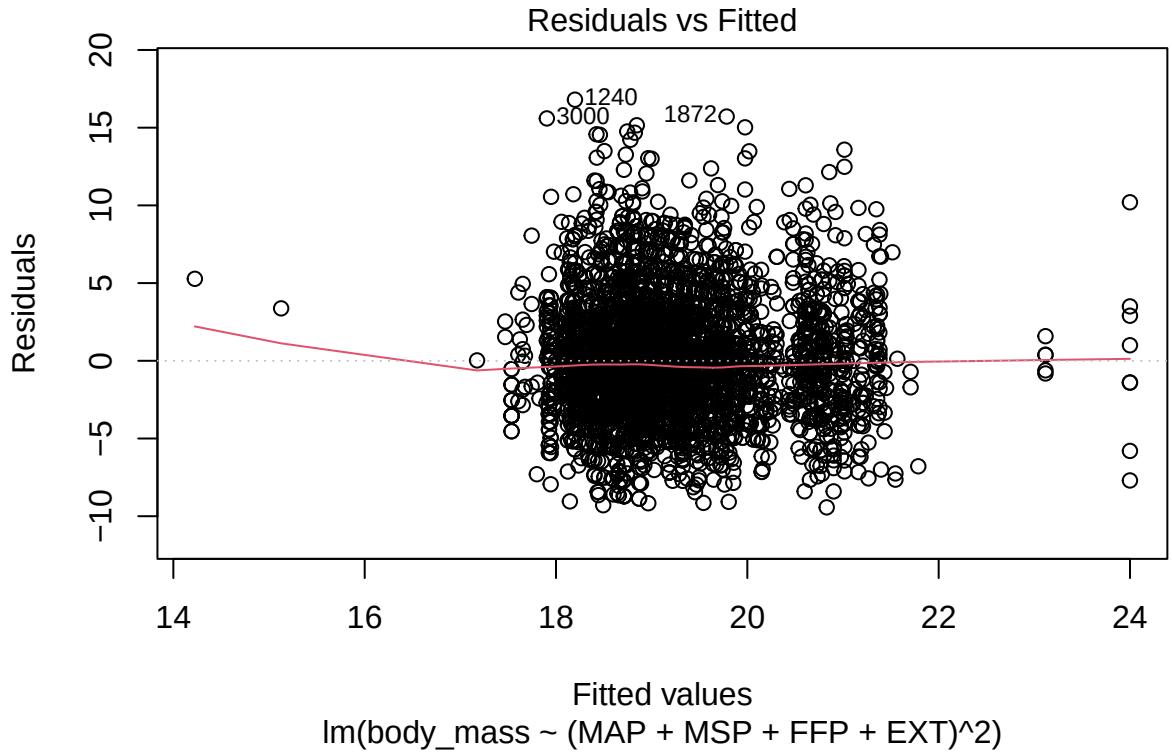
The computed RSME:

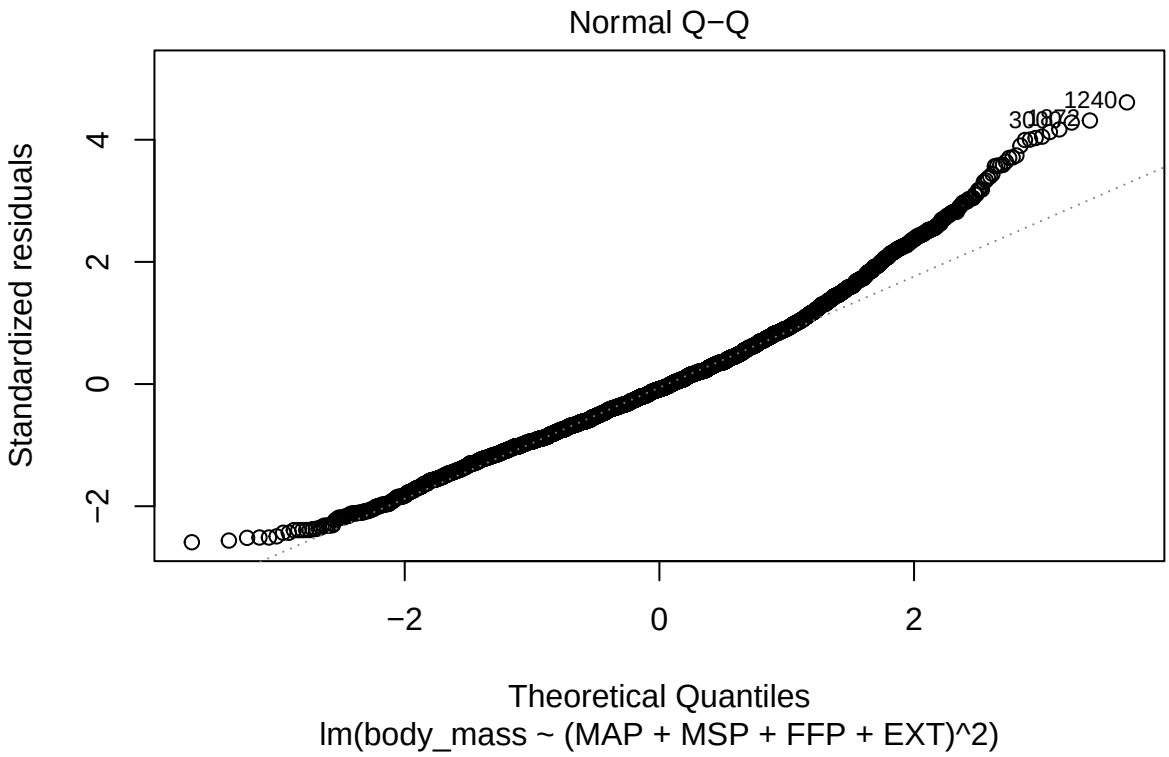
```
sqrt(mean(m.bm$residuals ^ 2))
```

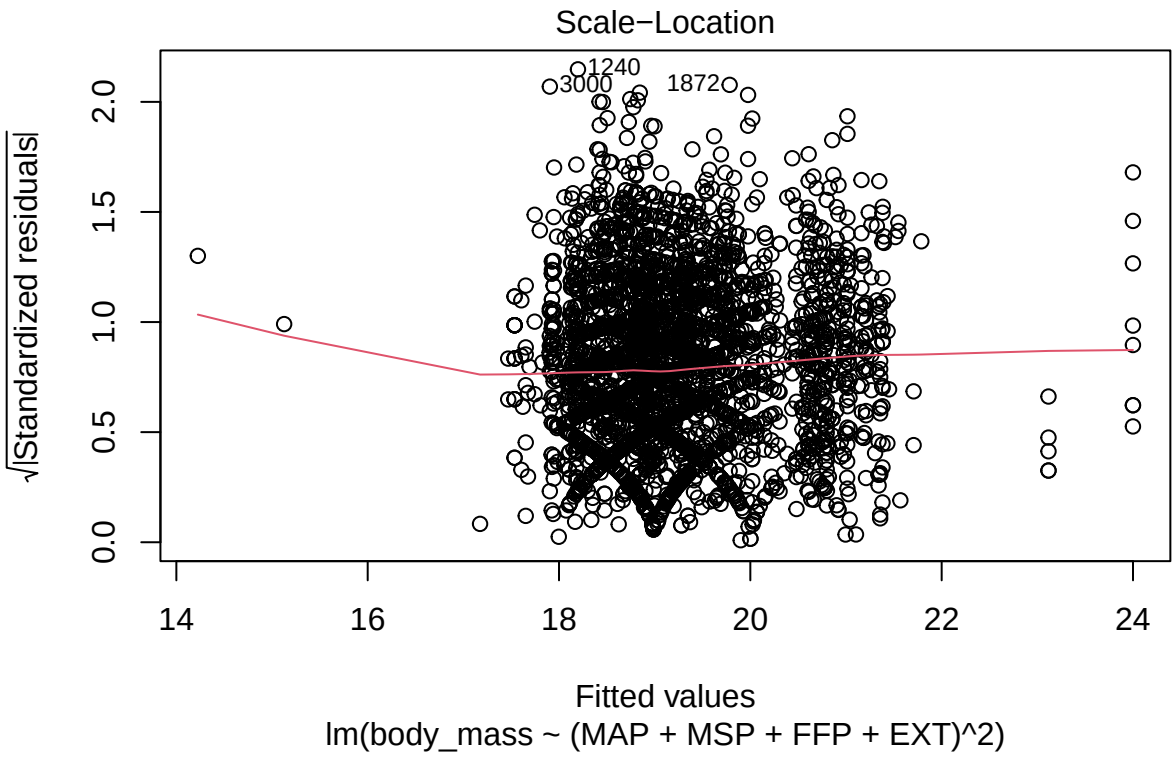
```
## [1] 3.640712
```

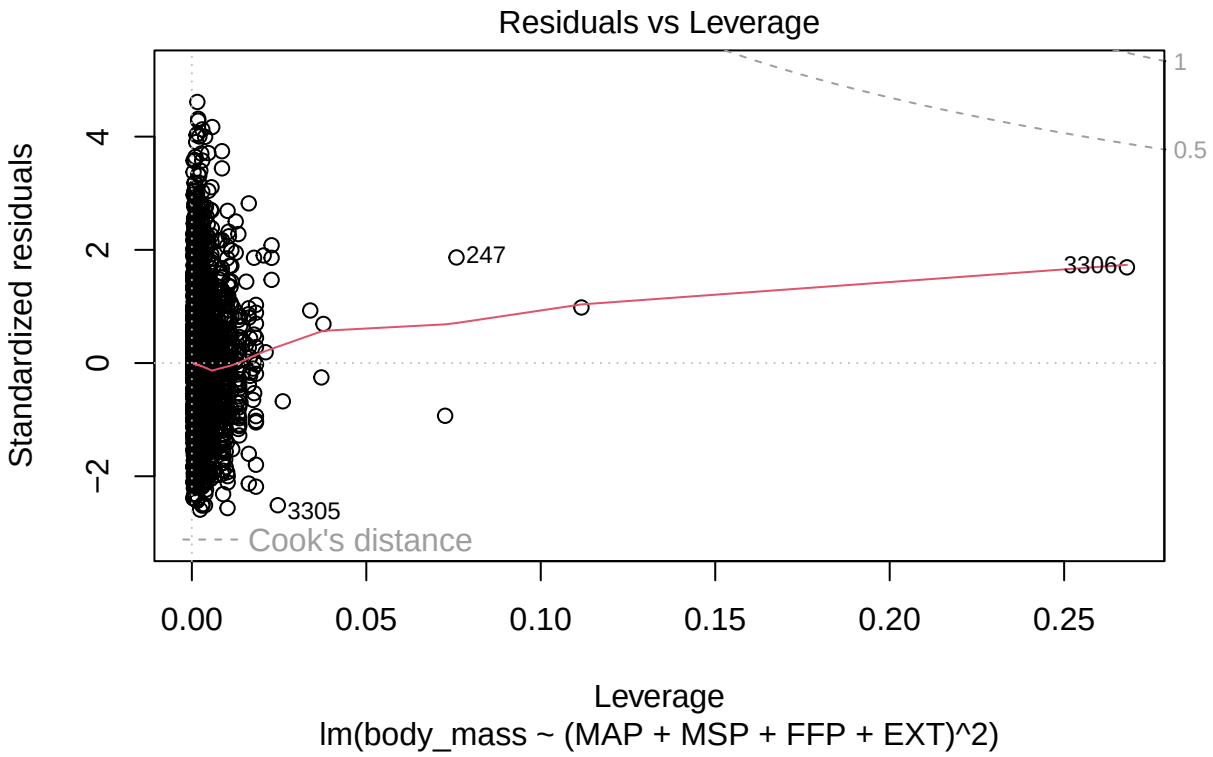
The visualization for the assumption inspection:

```
plot(m.bm)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(body_mass ~ (MAP + MSP + FFP + EXT)^2)
1240
−2
0
2
4


Scale−Location
1240
3000
1872
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
14
16
18
20
22
24
Fitted values
lm(body_mass ~ (MAP + MSP + FFP + EXT)^2)



As we can see above by the visual inspection, the homoscedasticity of residuals looks correct, however, looking at the qq-plot, we can see that normality is not there. There are no significant outliers measured by Cook's distance.

To sum up, it looks like the climate has very little to no impact on the body mass and total length of the mice. It should be noted that total length is more dependant on the climate ($R^2_{adj} = 0.1225$) than the body mass ($R^2_{adj} = 0.0474$).

5.2. Regression tree

```
df<-read.csv("mice_filled_all_values.csv") %>% dplyr::select(-c(sex_transformed,ecoregion1_transformed,s
df <- df[df$sp != "Peromyscus maniculatus", ]

climate.bm.df<-df %>% filter(lifestage=="AD") %>% dplyr::select(c(body_mass,TD,MAP,MSP,FFP,EXT))
dim(climate.bm.df)
```

```
## [1] 4512    6
```

```
set.seed (10)

splitIndex <- sample(1:nrow(climate.bm.df), size = 3/4 * nrow(climate.bm.df))
train.climate.bm.df <- climate.bm.df[splitIndex, ]
test.climate.bm.df <- climate.bm.df[-splitIndex, ]
```

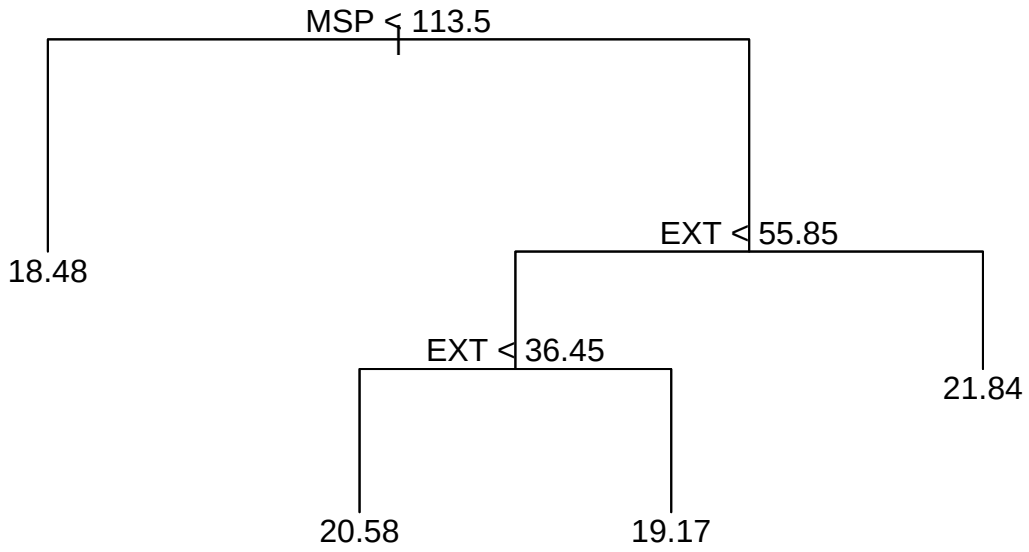
```
tree.mass.climate<-tree(body_mass ~. , data=train.climate.bm.df)

summary(tree.mass.climate)
```

```
##
## Regression tree:
## tree(formula = body_mass ~ ., data = train.climate.bm.df)
## Variables actually used in tree construction:
## [1] "MSP" "EXT"
## Number of terminal nodes:  4
## Residual mean deviance:  13.05 = 44100 / 3380
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -11.4800  -2.4810  -0.4815   0.0000   1.9220  15.8300
```

```
tree.mass.climate
```

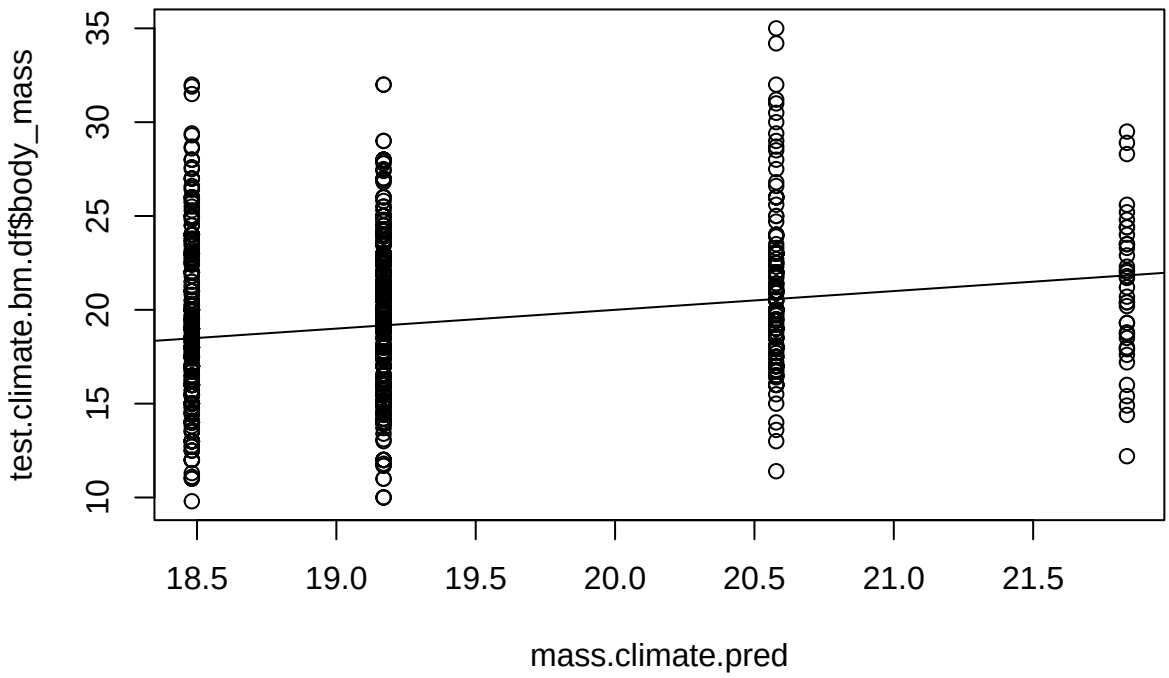
```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 3384 46620 19.23
##    2) MSP < 113.5 1269 16130 18.48 *
##    3) MSP > 113.5 2115 29350 19.68
##      6) EXT < 55.85 1989 26580 19.54
##       12) EXT < 36.45 521  6420 20.58 *
##       13) EXT > 36.45 1468 19400 19.17 *
##      7) EXT > 55.85 126  2146 21.84 *
```

```
plot(tree.mass.climate)
text(tree.mass.climate ,pretty =0)
```



The tree showed that the root node had 3384 observations, with a mean body mass of 19.23. The first split was based on MSP, with a threshold of 113.5. The left branch had 1269 observations with a mean body mass of 18.48, while the right branch had 2115 observations with a mean body mass of 19.68. The second split was based on EXT, with a threshold of 55.85. The left branch had 1989 observations with a mean body mass of 19.54, and it was further split based on EXT, with a threshold of 36.45. This left branch had 521 observations with a mean body mass of 20.58. The right branch had 1468 observations with a mean body mass of 19.17, and it was also further split based on EXT, but no further improvement in the model was obtained. The right branch of the second split had 126 observations with a mean body mass of 21.84.

```
mass.climate.pred<-predict(tree.mass.climate,test.climate.bm.df)
plot(mass.climate.pred,test.climate.bm.df$body_mass)
abline(0,1)
```

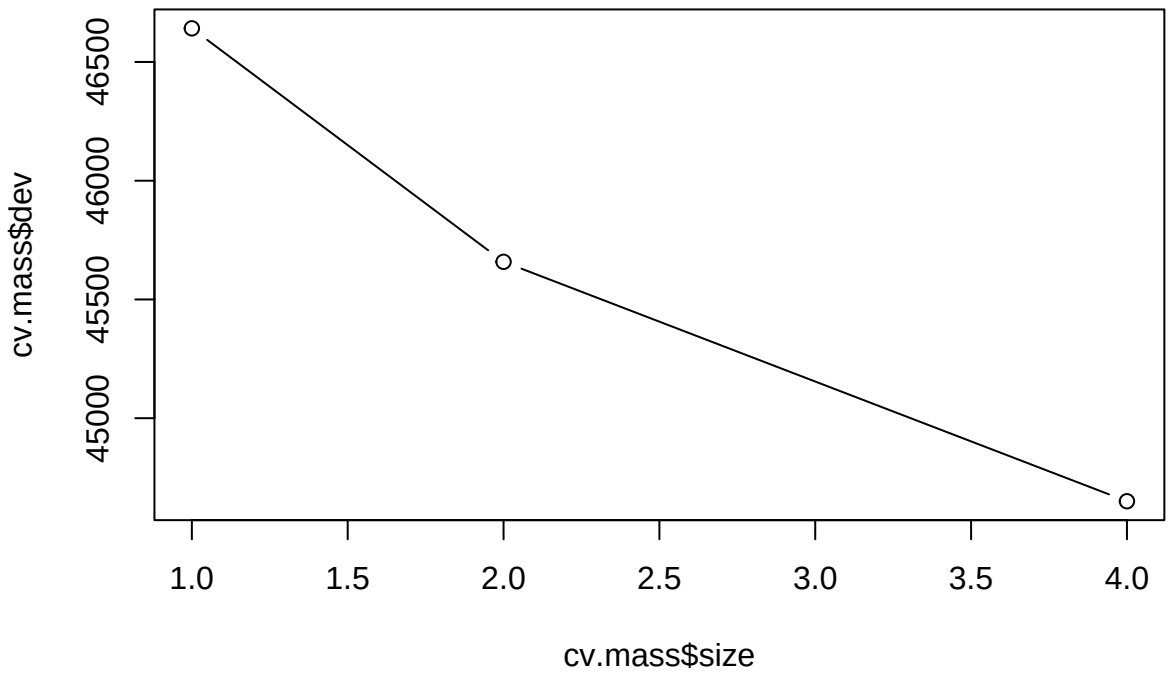



```
sqrt(mean((mass.climate.pred-test.climate.bm.df$body_mass)^2)) #RMSE
```

```
## [1] 3.774306
```

The model showed that MSP and EXT were the only variables used in the tree construction, and the number of terminal nodes was four. The residual mean deviance was 13.05, and the RMSE is 3.7743 indicating a good fit of the model to the data.

```
cv.mass<-cv.tree(tree.mass.climate)
plot(cv.mass$size,cv.mass$dev,type="b")
```



The plot of cross-validation error versus tree size for the body mass climate model shows no clear indication for pruning. The plot suggests that the tree is not too complex and has not overfit the training data, and therefore pruning may not be necessary.

```
climate.len.df<-df %>% filter(lifestage=="AD") %>% dplyr::select(c(total_length, TD, MAP, MSP, FFP, EXT
dim(climate.len.df)
```

```
## [1] 4512    6
```

```
set.seed (10)


splitIndex <- sample(1:nrow(climate.len.df), size = 3/4 * nrow(climate.len.df))
train.climate.len.df <- climate.len.df[splitIndex, ]
test.climate.len.df <- climate.len.df[-splitIndex, ]
```

```
tree.len.climate<-tree(total_length ~. , data=train.climate.len.df)

summary(tree.len.climate)
```

```
##
## Regression tree:
## tree(formula = total_length ~ ., data = train.climate.len.df)
## Variables actually used in tree construction:
## [1] "MAP" "EXT" "MSP" "TD"
```

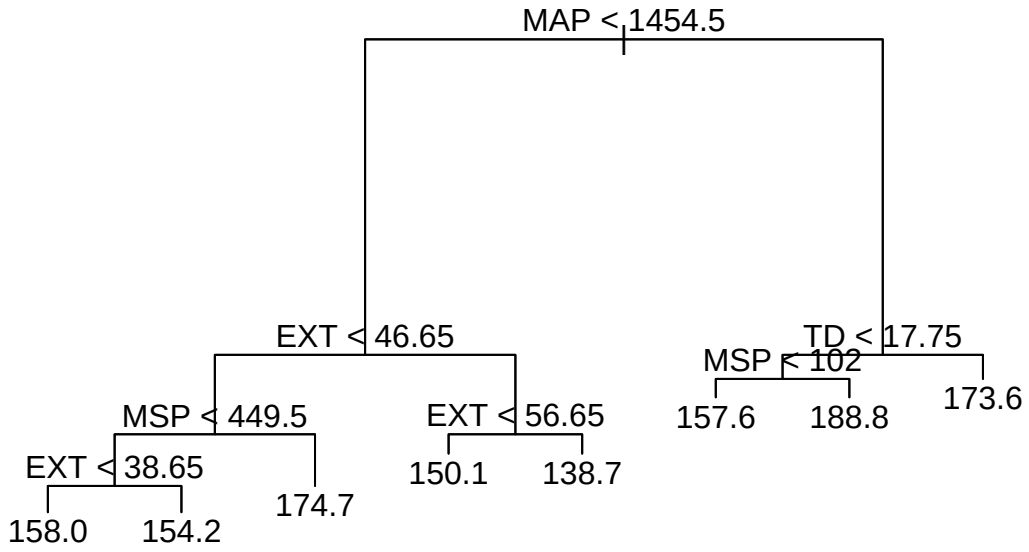
```
## Number of terminal nodes:  8
## Residual mean deviance:  125.6 = 424000 / 3376
## Distribution of residuals:
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -49.80000  -6.95300   0.04719   0.00000   7.04700  42.84000
```

```
tree.len.climate
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 3384 605000.0 157.1
##    2) MAP < 1454.5 3160 442300.0 155.6
##      4) EXT < 46.65 2668 339900.0 156.9
##        8) MSP < 449.5 2614 312000.0 156.5
##         16) EXT < 38.65 1600 155600.0 158.0 *
##         17) EXT > 38.65 1014 147500.0 154.2 *
##        9) MSP > 449.5 54  10490.0 174.7 *
##      5) EXT > 46.65 492  75430.0 148.8
##       10) EXT < 56.65 435  56150.0 150.1 *
##       11) EXT > 56.65 57  12720.0 138.7 *
##    3) MAP > 1454.5 224  55930.0 178.2
##      6) TD < 17.75 82  25190.0 186.1
##       12) MSP < 102 7    143.7 157.6 *
##       13) MSP > 102 75  18800.0 188.8 *
##      7) TD > 17.75 142  22600.0 173.6 *
```

For the regression tree for total length, the variables used in the tree construction are MAP, EXT, MSP, and TD. The tree has 8 terminal nodes and a residual mean deviance of 125.6, which is the sum of squared deviations of the observed values from the predicted values. The distribution of residuals ranges from -49.8 to 42.84.

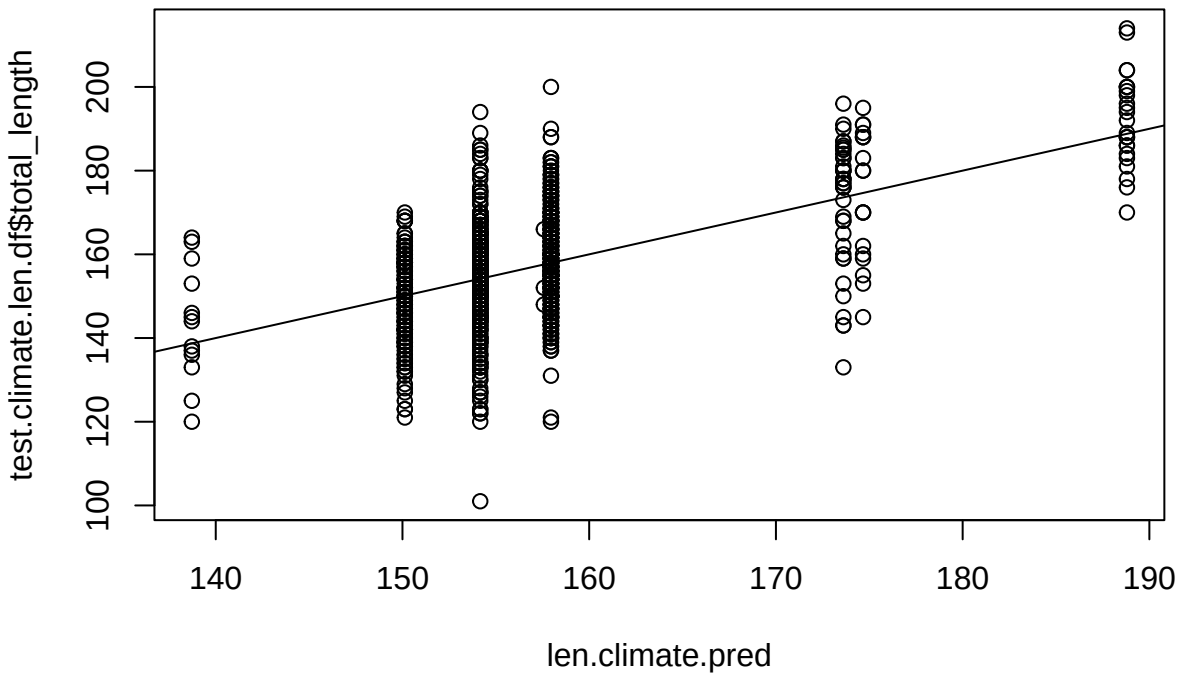
```
plot(tree.len.climate)
text(tree.len.climate ,pretty =0)
```



The first split of the tree is based on MAP, where observations with a MAP less than 1454.5 fall into node 2 and those with a MAP greater than or equal to 1454.5 fall into node 3. Node 2 is further split based on EXT, with observations with an EXT less than 46.65 falling into node 4 and those with an EXT greater than or equal to 46.65 falling into node 5. Node 4 is then split based on MSP, with observations with an MSP less than 449.5 falling into node 8, and those with an MSP greater than or equal to 449.5 falling into node 9. In node 8, observations with an EXT less than 38.65 fall into node 16, and those with an EXT greater than or equal to 38.65 fall into node 17. In node 9, observations fall into a single terminal node (node 9) based on the MSP split.

Node 5 is further split based on EXT, with observations with an EXT less than 56.65 falling into node 10, and those with an EXT greater than or equal to 56.65 falling into node 11. Node 3 is split based on TD, with observations with a TD less than 17.75 falling into node 6, and those with a TD greater than or equal to 17.75 falling into node 7. In node 6, observations with an MSP less than 102 fall into node 12, and those with an MSP greater than or equal to 102 fall into node 13. Node 7 is a single terminal node.

```
len.climate.pred<-predict(tree.len.climate,test.climate.len.df)
plot(len.climate.pred,test.climate.len.df$total_length)
abline(0,1)
```

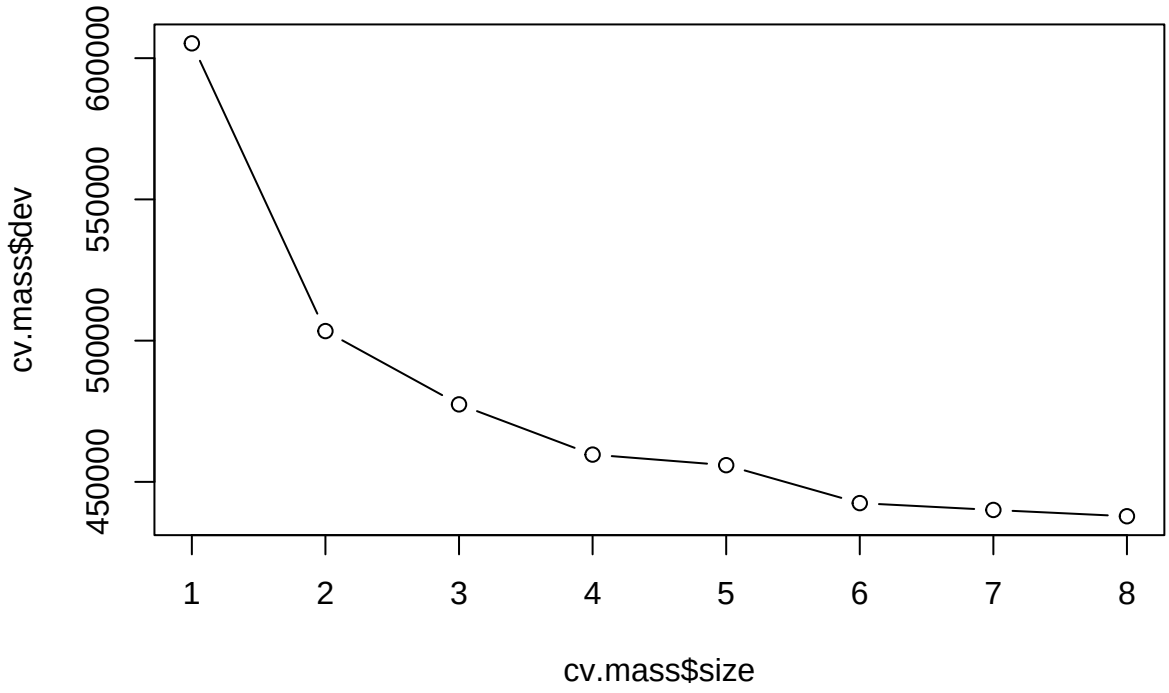



```
sqrt(mean((len.climate.pred-test.climate.len.df$total_length)^2)) #RMSE
```

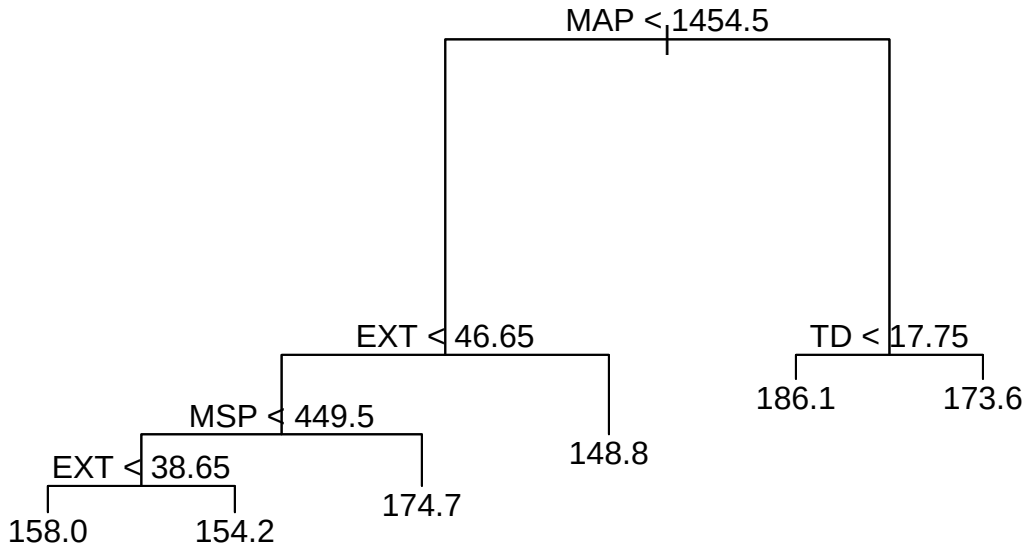
```
## [1] 11.35301
```

The root mean squared error (RMSE) for the regression tree on predicting total length using climate variables is 11.35301. A root mean squared error (RMSE) of 11.35301 indicates that the model has relatively high error in predicting total fish length based on the given climate variables.

```
cv.mass<-cv.tree(tree.len.climate)
plot(cv.mass$size,cv.mass$dev,type="b")
```



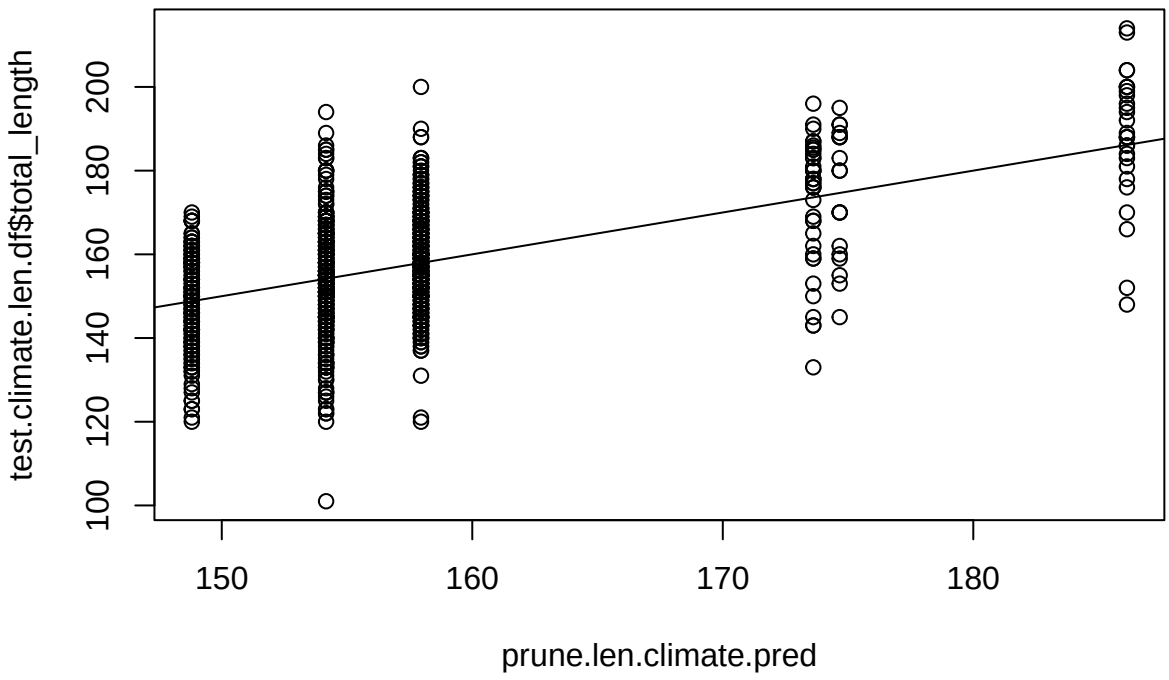

```
prune.length.climate=prune.tree(tree.len.climate,best=6)
plot(prune.length.climate)
text(prune.length.climate,pretty=0)
```




```
prune.length.climate
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 3384 605000 157.1
##    2) MAP < 1454.5 3160 442300 155.6
##      4) EXT < 46.65 2668 339900 156.9
##        8) MSP < 449.5 2614 312000 156.5
##         16) EXT < 38.65 1600 155600 158.0 *
##         17) EXT > 38.65 1014 147500 154.2 *
##        9) MSP > 449.5 54  10490 174.7 *
##      5) EXT > 46.65 492  75430 148.8 *
##    3) MAP > 1454.5 224  55930 178.2
##      6) TD < 17.75 82  25190 186.1 *
##      7) TD > 17.75 142  22600 173.6 *
```

```
prune.len.climate.pred<-predict(prune.length.climate,test.climate.len.df)
plot(prune.len.climate.pred,test.climate.len.df$total_length)
abline(0,1)
```




```
sqrt(mean((prune.len.climate.pred-test.climate.len.df$total_length)^2)) #RMSE
```

```
## [1] 11.4887
```

Regression Tree for Body Mass on Climate Data

```
set.seed (10)

df<-read.csv("mice_filled_all_values.csv") %>% dplyr::select(-c(sex_transformed,ecoregion1_transformed,s
df <- df[df$sp != "Peromyscus maniculatus", ]

climate.bm.df<-df %>% filter(lifestage=="AD") %>% dplyr::select(c(body_mass,TD,MAP,MSP,FFP,EXT))
dim(climate.bm.df)
```

```
## [1] 4512    6
```

```
set.seed (10)

splitIndex <- sample(1:nrow(climate.bm.df), size = 3/4 * nrow(climate.bm.df))
train.climate.bm.df <- climate.bm.df[splitIndex, ]
test.climate.bm.df <- climate.bm.df[-splitIndex, ]

# Specify the number of folds for cross-validation
```

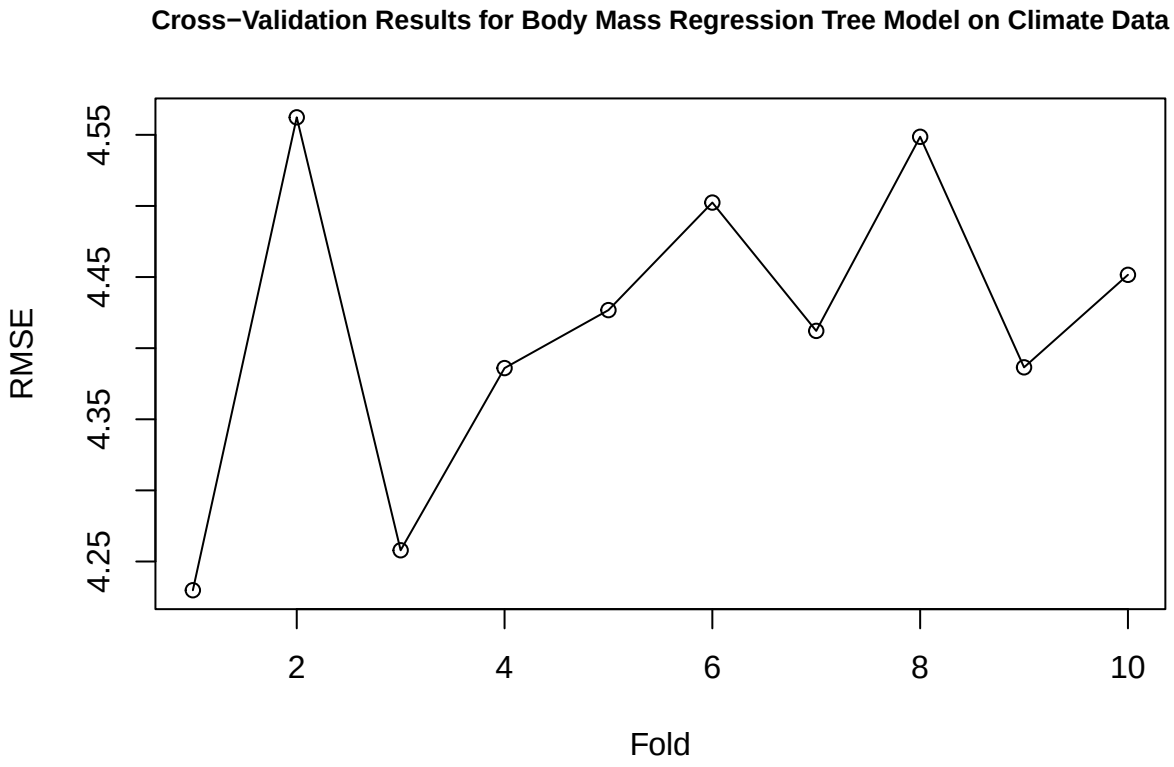
```r
k <- 10

folds <- createFolds(train.climate.bm.df$body_mass, k = k)

performance_metrics <- rep(0, k)

for (i in 1:k) {
  train_indices <- unlist(folds[-i])
  valid_indices <- folds[[i]]
  train_data <- train[train_indices, ]
  valid_data <- train[valid_indices, ]

  model <- tree.mass.climate

  predictions <- predict(model, newdata = valid_data)
  performance_metric <- sqrt(mean((predictions - valid_data$body_mass)^2))

  performance_metrics[i] <- performance_metric
}

average_performance <- mean(performance_metrics)
performance_metrics
```

```
##  [1] 4.229789 4.562297 4.257884 4.385992 4.426748 4.502379 4.412141 4.548629
##  [9] 4.386509 4.451559
```

```r
average_performance
```

```
## [1] 4.416393
```

```r
plot(performance_metrics, type = "o", xlab = "Fold", ylab = "RMSE", main = "Cross-Validation Results fo
```



We also conducted cross-validation for the regression tree model for body mass on the climate data. The results showed an average RMSE of 4.41, which is higher than the RMSE for the model on the original data.

The plot of the cross-validation error as a function of the tree size suggests that the optimal tree size is 6. Pruning the tree to this size results in the same tree as the unpruned tree. The RMSE of the pruned tree is 11.4887

In conclusion, the regression tree analysis conducted on the climate data to predict body mass and total length resulted in trees with 4 and 8 terminal nodes, respectively. The tree for body mass was pruned to 4 nodes, while the tree for total length was pruned to 6 nodes. The RMSE for the body mass prediction was 3.774306, indicating good predictive accuracy. However, the RMSE for total length prediction was 11.353, indicating relatively poor predictive accuracy. Therefore, while the regression tree model was effective in predicting body mass, it may not be the best model for predicting total length on the climate data.

5.3. Regression Tree for Predicting Total Length on Climate Data

```
k <- 10
set.seed (10)

folds <- createFolds(train.climate.len.df$total_length, k = k)

performance_metrics <- rep(0, k)

for (i in 1:k) {
  train_indices <- unlist(folds[-i])
```

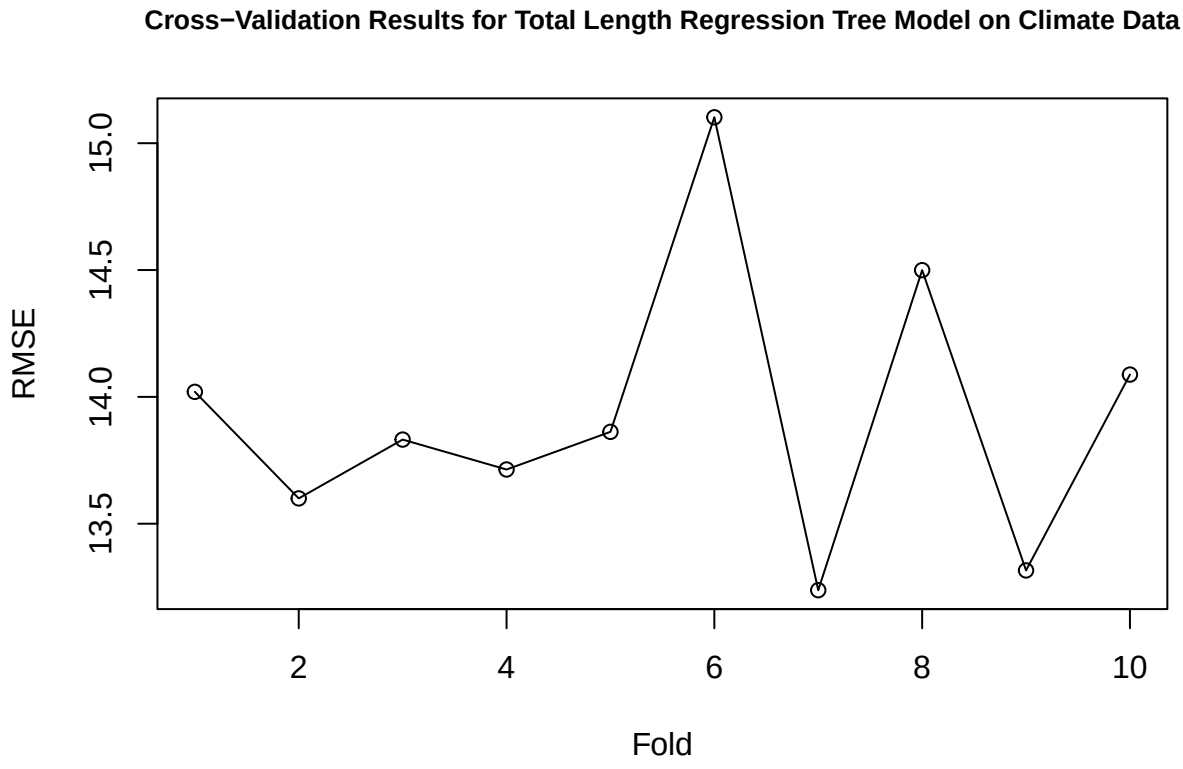
```
  valid_indices <- folds[[i]]
  train_data <- train[train_indices, ]
  valid_data <- train[valid_indices, ]

  model <- prune.length.climate

  predictions <- predict(model, newdata = valid_data)
  performance_metric <- sqrt(mean((predictions - valid_data$total_length)^2))

  performance_metrics[i] <- performance_metric
}

average_performance <- mean(performance_metrics)
performance_metrics
```

```
##  [1] 14.01979 13.59974 13.83171 13.71372 13.86220 15.10212 13.23798 14.49952
##  [9] 13.31586 14.08798
```

```
average_performance
```

```
## [1] 13.92706
```

```
plot(performance_metrics, type = "o", xlab = "Fold", ylab = "RMSE", main = "Cross-Validation Results fo
```



The regression tree model for predicting total length on climate data had an average performance metric of 13.93 across the 10 folds of cross-validation. This suggests that the model had a relatively high degree of error in its predictions of total length based on the climate data. It is important to note that this model did not include any length variables as predictors, only climate data. Therefore, it may be useful to compare the performance of this model to those including length variables as predictors.

5.5 Summary of the regression problems

This part of analysis presents the use of regression tree models to predict the body mass and total length of deer mice. The regression tree model for predicting body mass was based on the variables lifestage, total_length, and HB.Length. The 10-fold cross-validation average RMSE was 2.95 for the basic dataset and 4.41 for the basic dataset with climate data. This suggests that the basic dataset is the optimal choice for predicting deer mouse body mass, and that the inclusion of climate data has resulted in decreased performance.

The regression tree model for predicting total length was based on the variables tail_length and HB.Length. The 10-fold cross-validation average RMSE was 6.48 for the basic dataset and 13.93 for the basic dataset with climate data, and 9.53 for the basic dataset without body length and tail length variables. This indicates that the basic dataset is the optimal choice for predicting deer mouse total length, and that the inclusion of climate data significantly impacts model performance, while ignoring body length and tail length variables also results in decreased performance.

In the regression tree models predicting the body mass and total length of deer mice used in this analysis, HB.Length was frequently included in the models as important predictor variables. This suggests this variables plays a significant role in determining the body mass and total length of deer mice.

In conclusion, this part of the analysis demonstrates the use of regression tree models to predict deer mouse body mass and total length, and emphasizes the impact of selecting appropriate variables and datasets on model performance. Despite slightly worse performance compared to linear regression models, it's advisable to use regression trees because they are not dependent on any underlying assumptions.

---

6. Classification of the species.

The purpose of this part of analysis is to test out different classification methods and determine which gives best results. The categorical variable that will be used for the purpose of the classification are the subspecies of deer mice.

6.1 LDA & QDA (Author: Hao Su)

We loaded the data LDA and QDA and removed the variables with multicollinearity as well as time columns. On top of that, the observation marked as "Peromyscus maniculatus" was removed, because this specie is data that cannot be determined for specific sub-species, it affects the classification of sub-species.

```
library(dplyr)
df<-read.csv("mice_filled_all_values.csv")%>%dplyr::select(-c(sex_transformed,ecoregion1_transformed,sea

df <- df %>%dplyr::select(-c(HB.Length, MAT, MWMT, MCMT, TD, DD5,X.1,X,long,lat,decade,month,year))

df <- df[df$sp != "Peromyscus maniculatus", ]
head(df)
```

```
##   pop_density_4km2 season lifestage                                    sp    sex
## 1       4.31576252   fall     YOUNG    Peromyscus maniculatus abietorum   male
## 2       0.07396096   fall        AD Peromyscus maniculatus Wagner, 1845   male
## 3       0.00000000   fall        AD  Peromyscus maniculatus sonoriensis female
## 4       2.66981006   fall        AD      Peromyscus maniculatus bairdii   male
## 5       0.77556884   fall        AD     Peromyscus maniculatus gambelii female
## 6     157.70269780   fall        AD     Peromyscus maniculatus gambelii   male
##   body_mass tail_length total_length  MAP MSP FFP   EMT  EXT
## 1      18.5          79          151 1136 374 153 -32.7 44.1
## 2      21.0          72          164  131  18 275 -10.3 43.7
## 3      17.0          66          155  411  15 136 -24.5 37.5
## 4      15.5          54          124 1620 562 115 -31.4 36.5
## 5      13.0          67          145 1108  39 117 -31.3 37.1
## 6      17.3          72          152  955  29 307 -11.9 41.4
##                          ecoregion1
## 1       EASTERN TEMPERATE FORESTS
## 2        MEDITERRANEAN CALIFORNIA
## 3        MEDITERRANEAN CALIFORNIA
## 4 NORTHWESTERN FORESTED MOUNTAINS
## 5 NORTHWESTERN FORESTED MOUNTAINS
## 6        MEDITERRANEAN CALIFORNIA
```

When we did the QDA model, we found there are some species with very few observations, and QDA cannot deal with it, so for the same prediction comparison, we only kept subspecies with the observation count greater than 100. And the species "Peromyscus maniculatus abietorum" has some defects that make the model unworkable, so we delete it as well.

```
sp_count <- table(df$sp)

df <- df[df$sp %in% names(sp_count)[sp_count > 100], ]

df <- df[df$sp != "Peromyscus maniculatus abietorum", ]
table(df$sp)
```

```
##
##   Peromyscus maniculatus artemisiae      Peromyscus maniculatus bairdii
##                                 193                                 245
##     Peromyscus maniculatus gambelii       Peromyscus maniculatus luteus
##                                1772                                 513
## Peromyscus maniculatus nebrascensis      Peromyscus maniculatus rubidus
##                                 853                                 153
##      Peromyscus maniculatus rufinus  Peromyscus maniculatus sonoriensis
##                                 706                                1098
## Peromyscus maniculatus Wagner, 1845
##                                 617
```

And just keep the numeric variables, because LDA can only work well with numeric variables.

```
df2=df[,c("pop_density_4km2","sp","body_mass","tail_length","total_length","MAP","MSP","FFP","EMT")]
```

A training set (75%) and a test set (25%) were set up to verify the accuracy of the models and to determine which model is more accurate.

```
set.seed (2023)
idx=sample(1:nrow(df2),3/4*nrow(df2))
train=df2[idx,]
test=df2[-idx,]
```

Create LDA model by using all remaining columns.

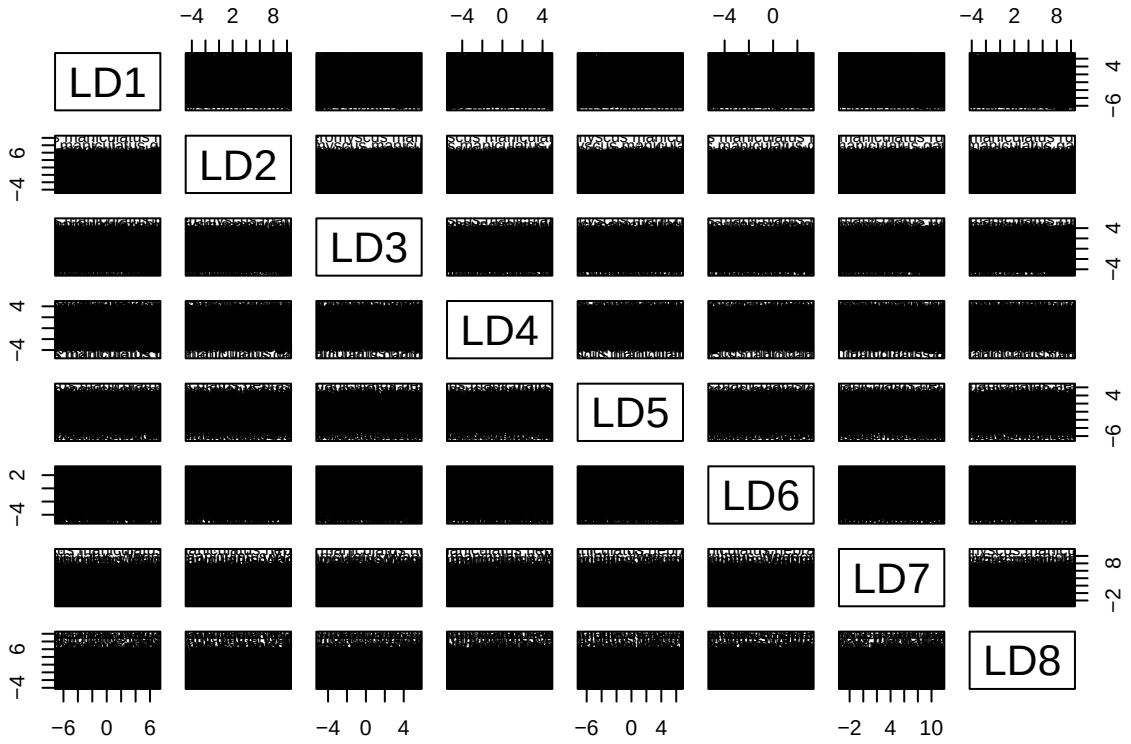
```
library(MASS)
lda.fit<-lda(sp~., data = train)
lda.fit
```

```
## Call:
## lda(sp ~ ., data = train)
##
## Prior probabilities of groups:
##   Peromyscus maniculatus artemisiae      Peromyscus maniculatus bairdii
##                          0.02970512                          0.04206418
##     Peromyscus maniculatus gambelii       Peromyscus maniculatus luteus
##                          0.28989592                          0.08304423
## Peromyscus maniculatus nebrascensis      Peromyscus maniculatus rubidus
##                          0.13768430                          0.02471813
##      Peromyscus maniculatus rufinus  Peromyscus maniculatus sonoriensis
##                          0.11426713                          0.18018213
## Peromyscus maniculatus Wagner, 1845
##                          0.09843886
##
## Group means:
##                                     pop_density_4km2 body_mass tail_length
## Peromyscus maniculatus artemisiae           1.823641  17.78686    77.21533
## Peromyscus maniculatus bairdii            141.662676  20.43041    60.16495
## Peromyscus maniculatus gambelii            12.193374  16.85329    68.92386
## Peromyscus maniculatus luteus               4.368032  18.54360    58.63708
## Peromyscus maniculatus nebrascensis        38.570959  17.22661    61.75512
## Peromyscus maniculatus rubidus             70.287466  17.32632    88.53947
## Peromyscus maniculatus rufinus             23.754080  18.11245    63.68027
## Peromyscus maniculatus sonoriensis          1.417175  17.03767    67.58917
## Peromyscus maniculatus Wagner, 1845        28.362730  17.69890    68.49449
##                                     total_length       MAP       MSP       FFP
## Peromyscus maniculatus artemisiae       164.3504  712.1533 236.24088  96.11679
## Peromyscus maniculatus bairdii          143.7887  962.5722 545.10825 191.63918
## Peromyscus maniculatus gambelii         151.3738  747.8362  89.72102 125.01346
## Peromyscus maniculatus luteus           150.8590  570.8668 396.45170 145.91906
## Peromyscus maniculatus nebrascensis     148.5622  346.7795 176.89134 129.41102
## Peromyscus maniculatus rubidus          172.2895 1691.4561 202.10526 219.63158
## Peromyscus maniculatus rufinus          149.9753  513.3169 215.82163 108.13093
## Peromyscus maniculatus sonoriensis      152.0903  376.5499  96.18893 137.96871
## Peromyscus maniculatus Wagner, 1845     151.6740  477.1652  55.03965 218.24670
##                                           EMT
## Peromyscus maniculatus artemisiae   -35.68759
## Peromyscus maniculatus bairdii      -27.93814
## Peromyscus maniculatus gambelii     -29.51107
## Peromyscus maniculatus luteus       -35.74883
## Peromyscus maniculatus nebrascensis -37.39496
```

```
## Peromyscus maniculatus rubidus      -15.67719
## Peromyscus maniculatus rufinus      -37.67173
## Peromyscus maniculatus sonoriensis  -32.24910
## Peromyscus maniculatus Wagner, 1845 -18.00000
##
## Coefficients of linear discriminants:
##                             LD1           LD2           LD3           LD4
## pop_density_4km2  0.0002103993  0.0001392598  3.160033e-04 -0.0009048840
## body_mass         0.0187930291  0.0105064515 -9.337391e-02 -0.1364069349
## tail_length      -0.0448407236  0.0217733508  8.273196e-02  0.0313756577
## total_length      0.0125085009 -0.0073572586  1.289983e-02  0.0545825926
## MAP              -0.0010098773  0.0011614399  1.273628e-03 -0.0003896146
## MSP               0.0100962918  0.0092993294  8.657672e-05 -0.0005351016
## FFP               0.0413564131 -0.0112565789  4.481913e-03  0.0252196343
## EMT              -0.3747583852  0.2017713175 -1.296705e-01 -0.1225662844
##                             LD5           LD6           LD7           LD8
## pop_density_4km2 -0.0021960523 -0.001683919  0.0045319561  0.004224242
## body_mass        -0.1722130643 -0.031935315  0.0774567210 -0.219875925
## tail_length      -0.2031014384  0.048793357 -0.0635761322 -0.025813785
## total_length      0.1453139818  0.010921220  0.0625601812  0.030405887
## MAP               0.0008735136 -0.002542953 -0.0001033232 -0.000672189
## MSP              -0.0007902371  0.005821908 -0.0012438643  0.001832784
## FFP              -0.0092064256 -0.022686034 -0.0066701806 -0.007973199
## EMT               0.0870037691  0.162240416  0.0372253434  0.061028592
##
## Proportion of trace:
##    LD1    LD2    LD3    LD4    LD5    LD6    LD7    LD8
## 0.7266 0.1935 0.0413 0.0214 0.0092 0.0075 0.0002 0.0002
```

Tried to draw the image of LDA, but it was so messy that we couldn't see any information.

```
plot(lda.fit)
```



Calculate the missclassfication rate.

```
class.pred<-predict(lda.fit,test)
t <- table(class.pred$class,test$sp)

missclassfication <-(sum(class.pred$class !=test$sp))/dim(test)[1]
missclassfication
```

```
## [1] 0.2802341
```

Use the 10-folds cross-validation to get the average accuracy.

```
library(caret)
```

```
model_fit1<-train(sp~., data=df2, trControl = trainControl(method = "cv", number=10), method='lda')
model_fit1$results[2]
```

```
##    Accuracy
## 1 0.7118714
```

The accuracy in LDA model is 0.7118714.

Create QDA model by using all remaining columns.

```
qda.fit<-qda(sp~., data = train)
qda.fit
```

```
## Call:
## qda(sp ~ ., data = train)
##
## Prior probabilities of groups:
##   Peromyscus maniculatus artemisiae      Peromyscus maniculatus bairdii
##                           0.02970512                           0.04206418
##     Peromyscus maniculatus gambelii       Peromyscus maniculatus luteus
##                           0.28989592                           0.08304423
## Peromyscus maniculatus nebrascensis      Peromyscus maniculatus rubidus
##                           0.13768430                           0.02471813
##      Peromyscus maniculatus rufinus  Peromyscus maniculatus sonoriensis
##                           0.11426713                           0.18018213
## Peromyscus maniculatus Wagner, 1845
##                           0.09843886
##
## Group means:
##                                     pop_density_4km2 body_mass tail_length
## Peromyscus maniculatus artemisiae           1.823641  17.78686    77.21533
## Peromyscus maniculatus bairdii            141.662676  20.43041    60.16495
## Peromyscus maniculatus gambelii            12.193374  16.85329    68.92386
## Peromyscus maniculatus luteus               4.368032  18.54360    58.63708
## Peromyscus maniculatus nebrascensis        38.570959  17.22661    61.75512
## Peromyscus maniculatus rubidus             70.287466  17.32632    88.53947
## Peromyscus maniculatus rufinus             23.754080  18.11245    63.68027
## Peromyscus maniculatus sonoriensis          1.417175  17.03767    67.58917
## Peromyscus maniculatus Wagner, 1845        28.362730  17.69890    68.49449
##                                     total_length       MAP       MSP       FFP
## Peromyscus maniculatus artemisiae       164.3504  712.1533 236.24088  96.11679
## Peromyscus maniculatus bairdii          143.7887  962.5722 545.10825 191.63918
## Peromyscus maniculatus gambelii         151.3738  747.8362  89.72102 125.01346
## Peromyscus maniculatus luteus           150.8590  570.8668 396.45170 145.91906
## Peromyscus maniculatus nebrascensis     148.5622  346.7795 176.89134 129.41102
## Peromyscus maniculatus rubidus          172.2895 1691.4561 202.10526 219.63158
## Peromyscus maniculatus rufinus          149.9753  513.3169 215.82163 108.13093
## Peromyscus maniculatus sonoriensis      152.0903  376.5499  96.18893 137.96871
## Peromyscus maniculatus Wagner, 1845     151.6740  477.1652  55.03965 218.24670
##                                           EMT
## Peromyscus maniculatus artemisiae   -35.68759
## Peromyscus maniculatus bairdii      -27.93814
## Peromyscus maniculatus gambelii     -29.51107
## Peromyscus maniculatus luteus       -35.74883
## Peromyscus maniculatus nebrascensis -37.39496
## Peromyscus maniculatus rubidus      -15.67719
## Peromyscus maniculatus rufinus      -37.67173
## Peromyscus maniculatus sonoriensis  -32.24910
## Peromyscus maniculatus Wagner, 1845 -18.00000
```

Calculate the missclassfication rate.

```
class.pred<-predict(qda.fit,test)
t <- table(class.pred$class,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL

t
```

```
##
##         [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##    [1,]   51    0   27    0    0    0    8   30   10
##    [2,]    0   47    0    1    0    0    1    0    1
##    [3,]    0    0  311    0    0    0    0    5   38
##    [4,]    0    0    0  122    5    0    5    0    0
##    [5,]    0    1    0    4  186    0   31   11    0
##    [6,]    0    0    0    0    0   37    0    0    2
##    [7,]    1    3    0    2    9    0   49    0    0
##    [8,]    4    0   81    1   18    0   85  211   11
##    [9,]    0    0   16    0    0    2    0   10  101
```

```
missclassfication <-(sum(class.pred$class !=test$sp))/dim(test)[1]
missclassfication
```

```
## [1] 0.2750325
```

Use the 10-folds cross-validation to get the average accuracy.

```
model_fit2<-train(sp~., data=df2, trControl = trainControl(method = "cv", number=10), method='qda')
model_fit2$results[2]
```

```
##    Accuracy
## 1 0.7099238
```

The accuracy in LDA model is 0.7099238 which is less than 0.7118714 in LDA model.

Test of Multivariate Normality of LDA and QDA

$$H_0 : \text{The data follow normal distribution}$$
$$H_a \ : \text{The data do NOT follow normal distribution}$$

See what species we have. And do the test one by one.

```
names <-unique(df2$sp)
names
```

```
## [1] "Peromyscus maniculatus Wagner, 1845" "Peromyscus maniculatus sonoriensis"
## [3] "Peromyscus maniculatus bairdii"      "Peromyscus maniculatus gambelii"
## [5] "Peromyscus maniculatus artemisiae"   "Peromyscus maniculatus rufinus"
## [7] "Peromyscus maniculatus rubidus"      "Peromyscus maniculatus nebrascensis"
## [9] "Peromyscus maniculatus luteus"
```

```
library(energy)
```

```
##
## Attaching package: 'energy'
```

```
## The following objects are masked from 'package:Rfast':
##
##     bcdcor, dcor, dcor.ttest, dcov, edist
```

```
tmp.df<-df2 %>% filter(sp == "Peromyscus maniculatus luteus") %>% dplyr::select(-sp)
#tmp.df
mvnorm.etest(tmp.df, R=100)
```

```
##
##  Energy test of multivariate normality: estimated parameters
##
## data:  x, sample size 513, dimension 8, replicates 100
## E-statistic = 38.35, p-value < 2.2e-16
```

From the energy normality results above, we can see all the numerical predictors has the P-value less than 0.05, indicating the Normality assumption are not met.

Test of the Equality Variance

$$H_0 : \text{the data have equal variance}$$
$$H_a \ : \text{the data do NOT have equal variance}$$

See what columns we have. And do the test one by one.

```
names(df2)
```

```
## [1] "pop_density_4km2" "sp"               "body_mass"        "tail_length"
## [5] "total_length"     "MAP"              "MSP"              "FFP"
## [9] "EMT"
```

```
library(car)
leveneTest(EMT~factor(sp) , data = df2)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value    Pr(>F)
## group    8  168.69 < 2.2e-16 ***
##       6141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From the Levene's test results above, we can see all the numerical predictors has the P-value less than 0.05, indicating the equality of variance assumption are not met.

However, even if the assumptions are not met, our LDA and QDA model have been evaluated by cross-validation with k=10, and all the 10 folds have accuracy at around 70%, indicating our LDA and QDA model is stable and with good performance.

6.2. Tree classification (Author: Hao Su)

In the classification tree part, we only removed the useless columns, time related columns and the "Peromyscus maniculatus" and "Peromyscus maniculatus abietorum" species. And also for the same comparision, we chose the total number of specie greater than 100.

```
library(dplyr)
df<-read.csv("mice_filled_all_values.csv")%>%dplyr::select(-c(sex_transformed,ecoregion1_transformed,sea

df <- df %>%dplyr::select(-c(X.1,X,long,lat,decade,month,year))
df <- df[df$sp != "Peromyscus maniculatus", ]
#head(df)

sp_count <- table(df$sp)

df <- df[df$sp %in% names(sp_count)[sp_count > 100], ]

df <- df[df$sp != "Peromyscus maniculatus abietorum", ]
table(df$sp)
```

```
##
##   Peromyscus maniculatus artemisiae      Peromyscus maniculatus bairdii
##                                 193                                 245
##     Peromyscus maniculatus gambelii       Peromyscus maniculatus luteus
##                                1772                                 513
## Peromyscus maniculatus nebrascensis      Peromyscus maniculatus rubidus
##                                 853                                 153
##      Peromyscus maniculatus rufinus  Peromyscus maniculatus sonoriensis
##                                 706                                1098
## Peromyscus maniculatus Wagner, 1845
##                                 617
```

In this step, the categorical variable "ecoregion1" in the data was transformed into a new variable with initials. The original "ecoregion1" had nine unique values, each representing a different type of ecoregion. To simplify the data and make it easier to analyze, each of these values was transformed into a set of initials. For example, "EASTERN TEMPERATE FORESTS" was transformed into "ETF". This transformation allowed for easier manipulation and analysis of the data, as well as reducing the complexity of the model. By transforming the categorical variable into a new variable with initials, the data became more streamlined and manageable for further analysis.

```
unique(df$ecoregion1)
```

```
## [1] "MEDITERRANEAN CALIFORNIA"        "NORTHWESTERN FORESTED MOUNTAINS"
## [3] "NORTH AMERICAN DESERTS"          "MARINE WEST COAST FOREST"
## [5] "GREAT PLAINS"                    "EASTERN TEMPERATE FORESTS"
## [7] "TEMPERATE SIERRAS"
```

```
df$ecoregion1 <- gsub("EASTERN TEMPERATE FORESTS", "ETF", df$ecoregion1)
df$ecoregion1 <- gsub("MEDITERRANEAN CALIFORNIA", "MC", df$ecoregion1)
df$ecoregion1 <- gsub("NORTHWESTERN FORESTED MOUNTAINS", "NWFM", df$ecoregion1)
df$ecoregion1 <- gsub("NORTH AMERICAN DESERTS", "NAD", df$ecoregion1)
df$ecoregion1 <- gsub("MARINE WEST COAST FOREST", "MWCF", df$ecoregion1)
```

```
df$ecoregion1 <- gsub("GREAT PLAINS", "GP", df$ecoregion1)
df$ecoregion1 <- gsub("SOUTHERN SEMIARID HIGHLANDS", "SSH", df$ecoregion1)
df$ecoregion1 <- gsub("NORTHERN FORESTS", "NF", df$ecoregion1)
df$ecoregion1 <- gsub("TEMPERATE SIERRAS", "TS", df$ecoregion1)
unique(df$ecoregion1)
```

```
## [1] "MC"   "NWFM" "NAD"  "MWCF" "GP"   "ETF"  "TS"
```

Resolved errors, reduced misclassification rate

```
dim(df)
```

```
## [1] 6150   20
```

```
# Convert variables to factors
df$ecoregion1 <- as.factor(df$ecoregion1)
df$sp <- as.factor(df$sp)
df$sex <- as.factor(df$sex)
df$lifestage <- as.factor(df$lifestage)
df$season <- as.factor(df$season)

# Check levels of the factors
levels(df$ecoregion1)
```

```
## [1] "ETF"  "GP"   "MC"   "MWCF" "NAD"  "NWFM" "TS"
```

```
levels(df$sp)
```

```
## [1] "Peromyscus maniculatus artemisiae"   "Peromyscus maniculatus bairdii"
## [3] "Peromyscus maniculatus gambelii"     "Peromyscus maniculatus luteus"
## [5] "Peromyscus maniculatus nebrascensis" "Peromyscus maniculatus rubidus"
## [7] "Peromyscus maniculatus rufinus"      "Peromyscus maniculatus sonoriensis"
## [9] "Peromyscus maniculatus Wagner, 1845"
```

```
levels(df$sex)
```

```
## [1] "female" "male"
```

```
levels(df$lifestage)
```

```
## [1] "AD"    "SUBAD" "YOUNG"
```

```
levels(df$season)
```

```
## [1] "fall"   "spring" "summer" "winter"
```

```
# Remove redundant or not meaningful levels
df$ecoregion1 <- droplevels(df$ecoregion1)
df$sp <- droplevels(df$sp)
df$sex <- droplevels(df$sex)
df$lifestage <- droplevels(df$lifestage)
df$season <- droplevels(df$season)
dim(df)
```

```
## [1] 6150   20
```

A training set (75%) and a test set (25%) were set up to verify the accuracy of the models and to determine which model is more accurate.

```
set.seed (10)
idx=sample(1:nrow(df),3/4*nrow(df))
train=df[idx,]
test=df[-idx,]
```

Set the 10-folds cross-validation set.

```
library(caret)
library(MASS)
set.seed(10)
folds<-createFolds(df$sp, k=10)
```

```
library(tree)

tree.class<-tree(factor(sp)~., train)
summary(tree.class)
```

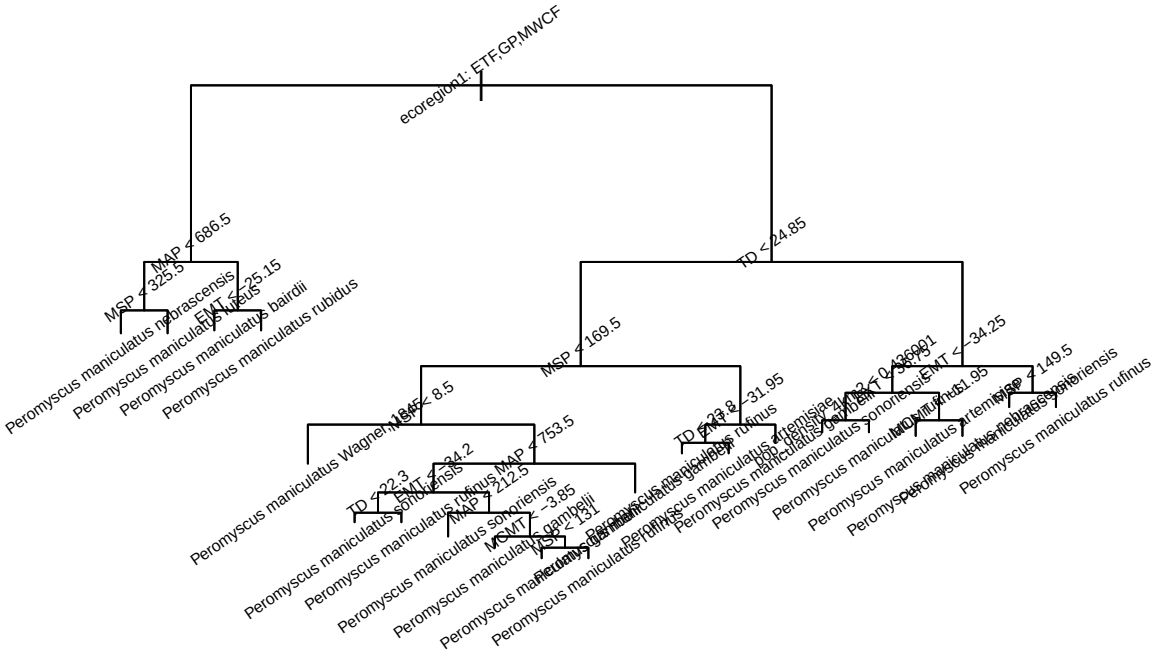
```
##
## Classification tree:
## tree(formula = factor(sp) ~ ., data = train)
## Variables actually used in tree construction:
## [1] "ecoregion1"       "MAP"              "MSP"              "EMT"
## [5] "TD"               "MCMT"             "EXT"              "pop_density_4km2"
## Number of terminal nodes:  21
## Residual mean deviance:  0.9653 = 4431 / 4591
## Misclassification error rate: 0.1581 = 729 / 4612
```

```
tree.class
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##   1) root 4612 18020.00 Peromyscus maniculatus gambelii ( 0.0299219 0.0390286 0.2883781 0.0810928 0.
##     2) ecoregion1: ETF,GP,MWCF 798  2105.00 Peromyscus maniculatus luteus ( 0.0000000 0.2243108 0.000
##       4) MAP < 686.5 507   708.70 Peromyscus maniculatus luteus ( 0.0000000 0.0118343 0.0000000 0.727
##         8) MSP < 325.5 122    95.75 Peromyscus maniculatus nebrascensis ( 0.0000000 0.0081967 0.00000
##         9) MSP > 325.5 385   166.70 Peromyscus maniculatus luteus ( 0.0000000 0.0129870 0.0000000 0.9
##       5) MAP > 686.5 291   454.50 Peromyscus maniculatus bairdii ( 0.0000000 0.5945017 0.0000000 0.0
```

```
##        10) EMT < -25.15 178    45.58 Peromyscus maniculatus bairdii ( 0.0000000 0.9719101 0.0000000 0
##        11) EMT > -25.15 113    20.10 Peromyscus maniculatus rubidus ( 0.0000000 0.0000000 0.0000000 0
##     3) ecoregion1: MC,NAD,NWFM,TS 3814 12480.00 Peromyscus maniculatus gambelii ( 0.0361825 0.000262
##       6) TD < 24.85 2669  7393.00 Peromyscus maniculatus gambelii ( 0.0243537 0.0003747 0.4837018 0.0
##        12) MSP < 169.5 2171  4858.00 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0.5642561
##          24) MSP < 8.5 217    67.03 Peromyscus maniculatus Wagner, 1845 ( 0.0000000 0.0000000 0.0046
##          25) MSP > 8.5 1954  4032.00 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0.6264074
##            50) MAP < 753.5 1307  3109.00 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0.472
##             100) EMT < -34.2 247   486.00 Peromyscus maniculatus sonoriensis ( 0.0000000 0.0000000 0
##               200) TD < 22.3 190   190.40 Peromyscus maniculatus sonoriensis ( 0.0000000 0.0000000 0
##               201) TD > 22.3 57   110.70 Peromyscus maniculatus rufinus ( 0.0000000 0.0000000 0.0000
##             101) EMT > -34.2 1060  2228.00 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0.5
##               202) MAP < 212.5 235   362.40 Peromyscus maniculatus sonoriensis ( 0.0000000 0.0000000
##               203) MAP > 212.5 825  1404.00 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0.
##                 406) MCMT < -3.85 283    13.29 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000
##                 407) MCMT > -3.85 542  1173.00 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000
##                   814) MSP < 131 481   845.20 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0
##                   815) MSP > 131 61   122.80 Peromyscus maniculatus rufinus ( 0.0000000 0.0000000 0.0
##            51) MAP > 753.5 647   362.40 Peromyscus maniculatus gambelii ( 0.0000000 0.0000000 0.9381
##        13) MSP > 169.5 498  1399.00 Peromyscus maniculatus rufinus ( 0.1305221 0.0020080 0.1325301 0
##          26) EMT < -31.95 378   764.10 Peromyscus maniculatus rufinus ( 0.1190476 0.0000000 0.002645
##            52) TD < 23.8 331   487.20 Peromyscus maniculatus rufinus ( 0.0422961 0.0000000 0.0030211
##            53) TD > 23.8 47    72.34 Peromyscus maniculatus artemisiae ( 0.6595745 0.0000000 0.000000
##          27) EMT > -31.95 120   280.10 Peromyscus maniculatus gambelii ( 0.1666667 0.0083333 0.541666
##       7) TD > 24.85 1145  3057.00 Peromyscus maniculatus nebrascensis ( 0.0637555 0.0000000 0.034061
##        14) EMT < -34.25 799  1724.00 Peromyscus maniculatus nebrascensis ( 0.0663329 0.0000000 0.0000
##          28) EXT < 36.75 230   444.00 Peromyscus maniculatus sonoriensis ( 0.0521739 0.0000000 0.0000
##            56) pop_density_4km2 < 0.436001 161   147.80 Peromyscus maniculatus sonoriensis ( 0.000000
##            57) pop_density_4km2 > 0.436001 69    63.76 Peromyscus maniculatus rufinus ( 0.1739130 0.0
##          29) EXT > 36.75 569   744.20 Peromyscus maniculatus nebrascensis ( 0.0720562 0.0000000 0.000
##            58) MCMT < -11.95 52    53.66 Peromyscus maniculatus artemisiae ( 0.7884615 0.0000000 0.00
##            59) MCMT > -11.95 517   387.70 Peromyscus maniculatus nebrascensis ( 0.0000000 0.0000000 0.
##        15) EMT > -34.25 346   815.60 Peromyscus maniculatus sonoriensis ( 0.0578035 0.0000000 0.1127
##          30) MSP < 149.5 261   433.60 Peromyscus maniculatus sonoriensis ( 0.0000000 0.0000000 0.1494
##          31) MSP > 149.5 85   103.10 Peromyscus maniculatus rufinus ( 0.2352941 0.0000000 0.0000000 0.
```

```
plot(tree.class)
text(tree.class, pretty=0,srt=35,cex=0.5)
```



Calculate the misclassification rate

```
# Make predictions on the test dataset using the pruned decision tree
tree.pred <- predict(tree.class, test, type = "class")

t=table(tree.pred,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL
t
```

```
##
## tree.pred [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##      [1,]   40    0    0    0    0    0    5    0    4
##      [2,]    0   61    0    4    0    0    0    0    0
##      [3,]    5    1  424    0    0    4    1   33   37
##      [4,]    0    1    0  135    3    0    0    0    0
##      [5,]    1    2    0    0  190    0    5    5    0
##      [6,]    0    0    0    0    0   29    0    0    0
##      [7,]    9    0    0    0   14    0  150   17    8
##      [8,]    0    0   16    0    6    0   15  214   32
##      [9,]    0    0    2    0    0    1    0    3   61
```

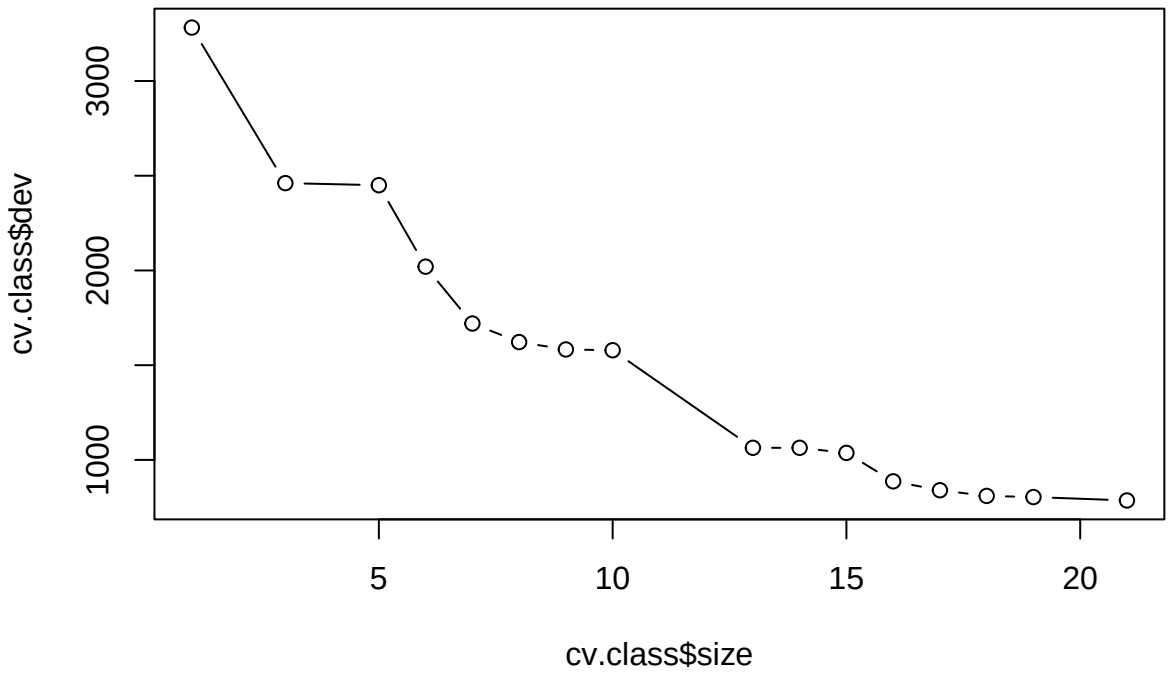
```
# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(tree.pred, levels = levels(test$sp))

# Calculate the misclassification rate
```

```
misclassification <- mean(prune.pred_factor != test$sp)

# Print the misclassification rate
cat("Misclassification rate:", misclassification, "\n")
```

```
## Misclassification rate: 0.1521456
```

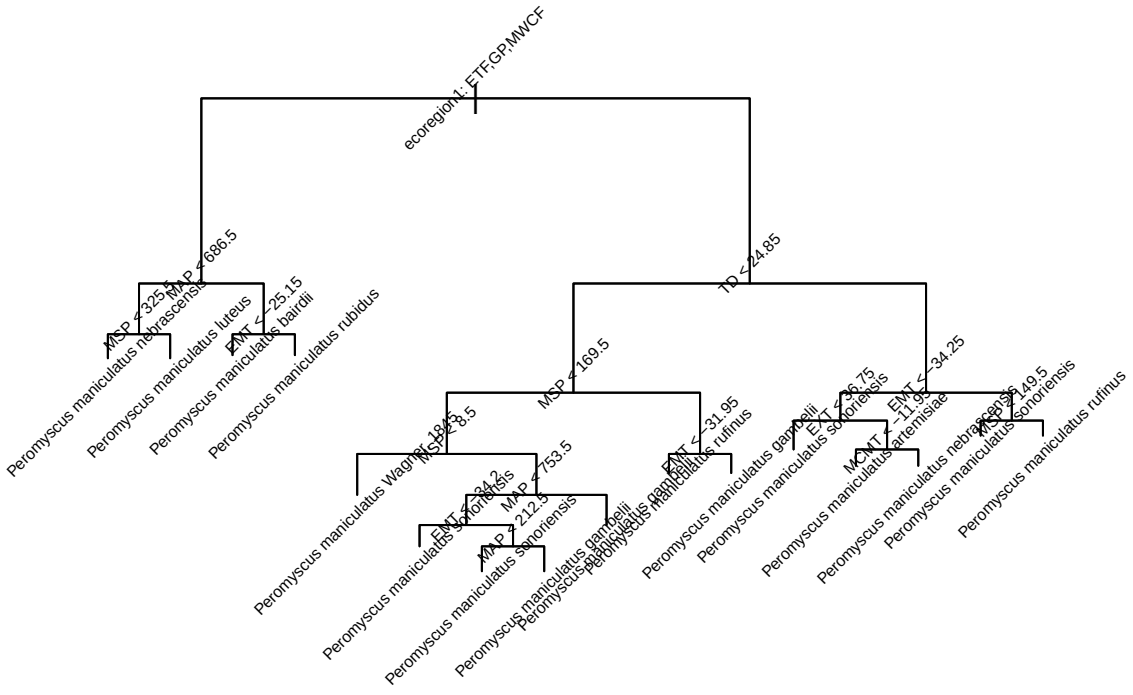
Prune the tree to prevent the over fitting.

```
set.seed(10)
cv.class<-cv.tree(tree.class, FUN = prune.misclass, K=10)
plot(cv.class$size, cv.class$dev,type="b")
```



We chose the best tree with 16 terminal nodes.

```
prune.class=prune.tree(tree.class,best=16)
plot(prune.class)
text(prune.class,pretty=0,srt=45,cex=0.5)
```



Calculate the misclassification rate.

```
prune.pred <- predict(prune.class, test, type = "class")
t=table(prune.pred,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL
t
```

```
##
## prune.pred [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##       [1,]   20    0    0    0    0    0    2    0    0
##       [2,]    0   61    0    4    0    0    0    0    0
##       [3,]    5    1  424    0    0    4   10   34   42
##       [4,]    0    1    0  135    3    0    0    0    0
##       [5,]    1    2    0    0  190    0    5    5    0
##       [6,]    0    0    0    0    0   29    0    0    0
##       [7,]   27    0    0    0   11    0  116   11    7
##       [8,]    2    0   16    0    9    0   43  219   32
##       [9,]    0    0    2    0    0    1    0    3   61
```

```
# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(prune.pred, levels = levels(test$sp))

# Calculate the misclassification rate
misclassification <- mean(prune.pred_factor != test$sp)
```

```
# Print the misclassification rate
cat("Misclassification rate:", misclassification, "\n")
```

```
## Misclassification rate: 0.1840052
```

Use 10-folds cross-validation to to get the average accuracy.

```
misclass_tree<-function(idx){
train=df[-idx,]
test=df[idx,]
tree.class<-tree(factor(sp)~., train)
prune.class=prune.tree(tree.class,best=16)
prune.pred <- predict(prune.class, test, type = "class")

# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(prune.pred, levels = levels(test$sp))

# Calculate the misclassification rate
misclassification <- mean(prune.pred_factor != test$sp)

return(misclassification)
}
mis_tree<-lapply(folds,misclass_tree)
print(1-mean(as.numeric(mis_tree)))
```

```
## [1] 0.7991918
```

The accuracy in classification tree model is 0.7991918.

Sex differentiation

We tried to predict sex, but the results of this tree show that there is no value in doing this.

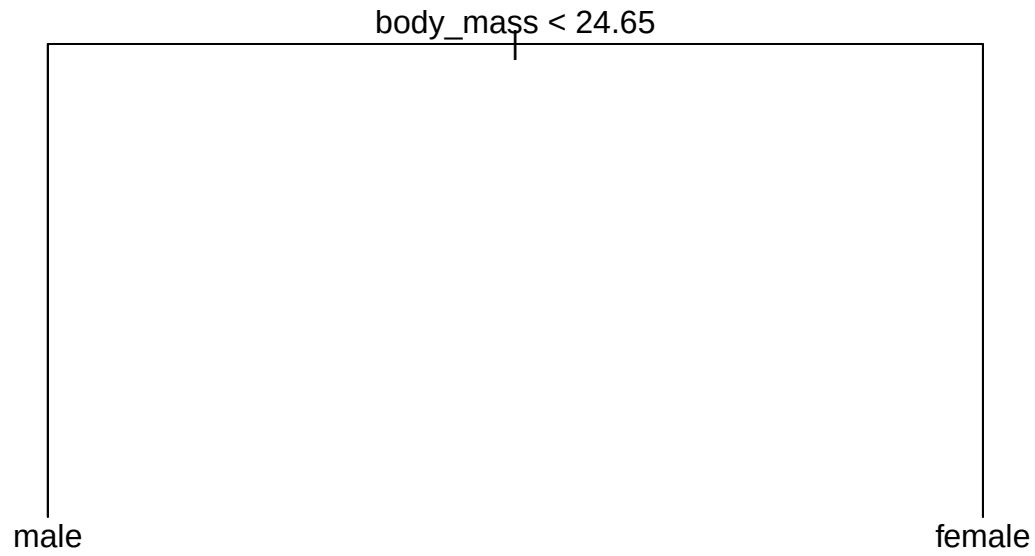
```
tree.class<-tree(factor(sex)~., train)
summary(tree.class)
```

```
##
## Classification tree:
## tree(formula = factor(sex) ~ ., data = train)
## Variables actually used in tree construction:
## [1] "body_mass"
## Number of terminal nodes:  2
## Residual mean deviance:  1.339 = 6171 / 4610
## Misclassification error rate: 0.3981 = 1836 / 4612
```

```
tree.class
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
## 1) root 4612 6296.0 male ( 0.4274 0.5726 )
##   2) body_mass < 24.65 4349 5879.0 male ( 0.4074 0.5926 ) *
##   3) body_mass > 24.65 263  291.9 female ( 0.7567 0.2433 ) *
```

```
plot(tree.class)
text(tree.class, pretty=0)
```



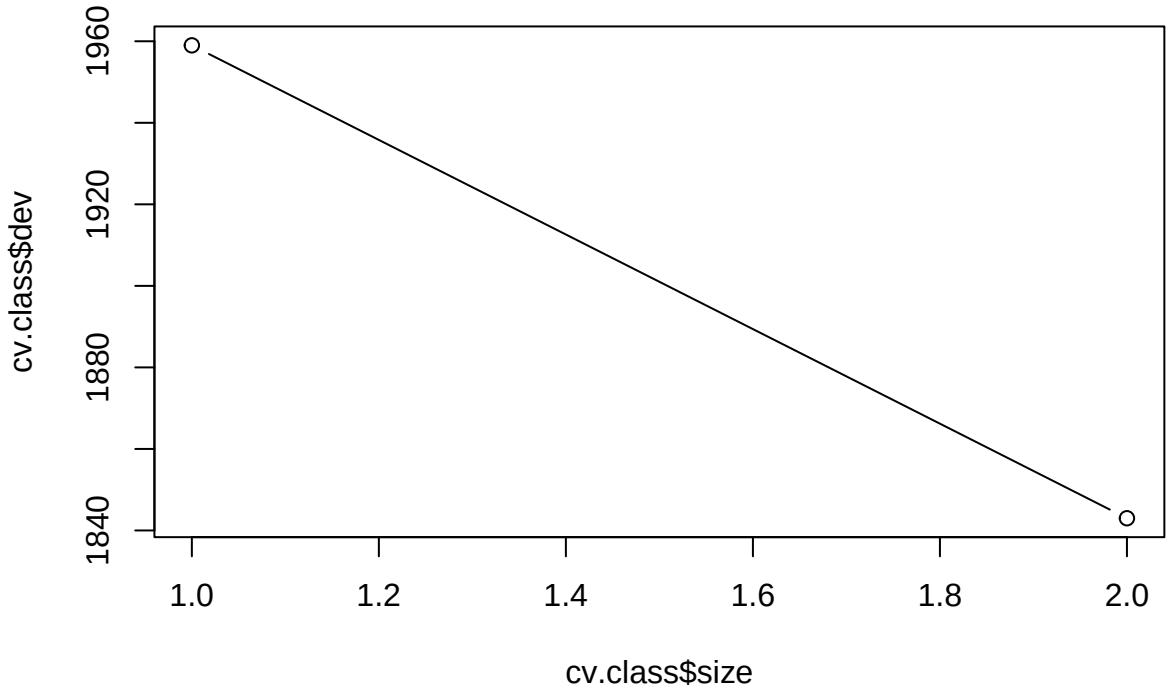

```
tree.pred<-predict(tree.class,test,type = "class")
table(tree.pred,test$sex)
```

```
##
## tree.pred female male
##    female     73   16
##    male      582  867
```

```
missclassfication <-(sum(tree.pred !=test$sex))/dim(test)[1]
missclassfication
```

```
## [1] 0.3888166
```

```
set.seed(10)
cv.class<-cv.tree(tree.class, FUN = prune.misclass, K=10)
plot(cv.class$size, cv.class$dev,type="b")
```



We divided the whole data into two parts, one is male and the other is female. And also created trees for these two sub data to compare which sex of the species can be predicted better. ## sp-male

```
male <- subset(df, sex == "male")
set.seed(10)
idx=sample(1:nrow(male),3/4*nrow(male))
train=male[idx,]
test=male[-idx,]
train$sp <- factor(train$sp)
test$sp <- factor(test$sp)
sapply(train, class)
```

```
## pop_density_4km2           season        lifestage               sp
##        "numeric"         "factor"         "factor"         "factor"
##              sex        body_mass     tail_length     total_length
##         "factor"        "numeric"        "numeric"        "numeric"
##        HB.Length              MAT             MWMT             MCMT
##        "numeric"        "numeric"        "numeric"        "numeric"
##               TD              MAP              MSP              DD5
##        "numeric"        "integer"        "integer"        "integer"
##              FFP              EMT              EXT       ecoregion1
##        "integer"        "numeric"        "numeric"         "factor"
```

```
sapply(test, class)
```

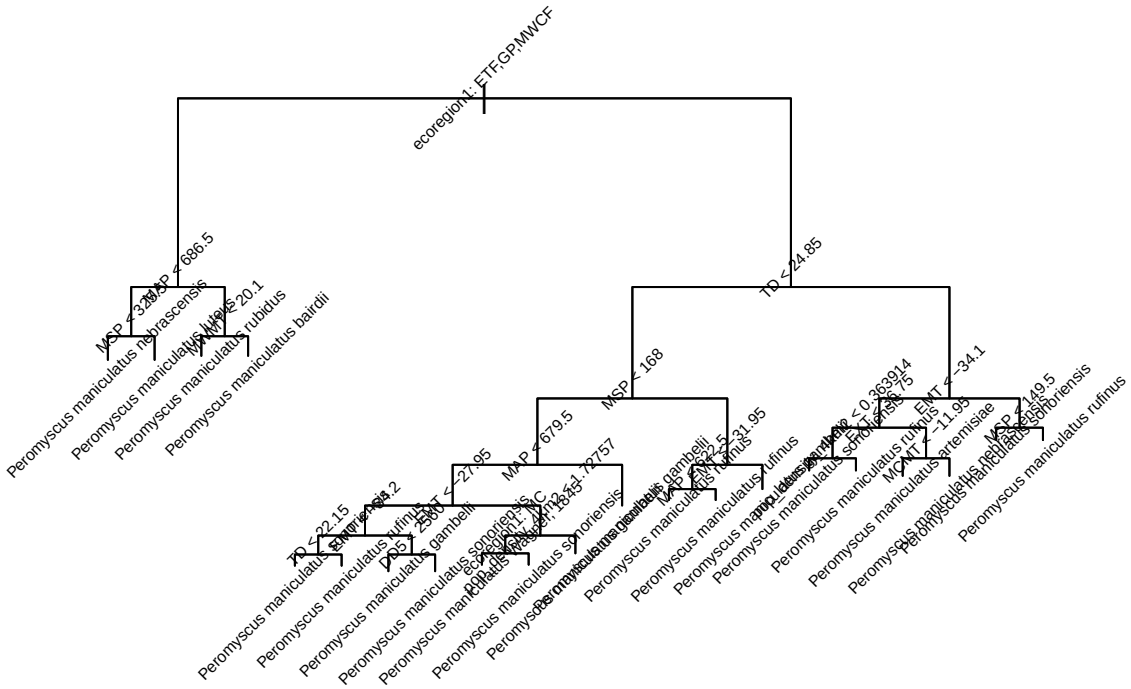
```
## pop_density_4km2           season        lifestage               sp
```

```
##        "numeric"         "factor"         "factor"         "factor"
##              sex        body_mass      tail_length     total_length
##         "factor"        "numeric"        "numeric"        "numeric"
##        HB.Length              MAT             MWMT             MCMT
##        "numeric"        "numeric"        "numeric"        "numeric"
##               TD              MAP              MSP              DD5
##        "numeric"        "integer"        "integer"        "integer"
##              FFP              EMT              EXT       ecoregion1
##        "integer"        "numeric"        "numeric"         "factor"
```

```
tree.class<-tree(factor(sp)~., train)
summary(tree.class)
```

```
##
## Classification tree:
## tree(formula = factor(sp) ~ ., data = train)
## Variables actually used in tree construction:
##  [1] "ecoregion1"       "MAP"              "MSP"              "MWMT"
##  [5] "TD"               "EMT"              "DD5"              "pop_density_4km2"
##  [9] "EXT"              "MCMT"
## Number of terminal nodes:  21
## Residual mean deviance:  0.9646 = 2529 / 2622
## Misclassification error rate: 0.1597 = 422 / 2643
```

```
plot(tree.class)
text(tree.class,pretty=0,srt=45,cex=0.5)
```



```r
# Make predictions on the test dataset
tree.pred <- predict(tree.class, newdata = test, type = "class")
t=table(tree.pred,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL
t
```

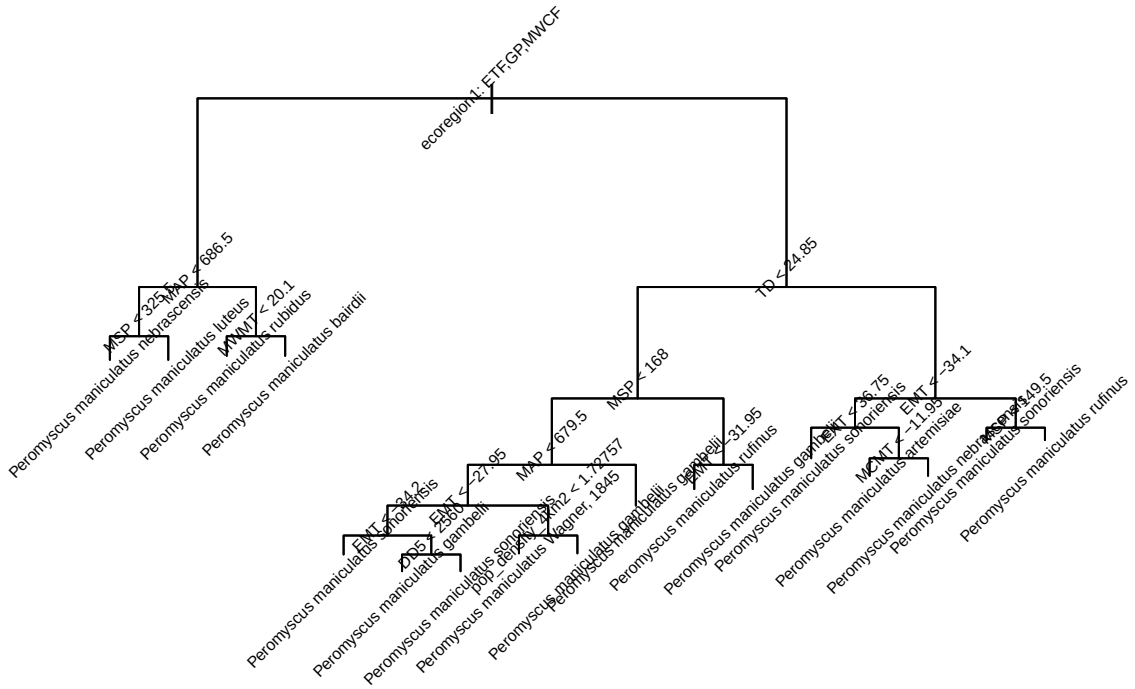
```
##
## tree.pred [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##      [1,]    6    0    0    0    0    0    4    0    0
##      [2,]    0   33    0    1    0    0    0    0    1
##      [3,]    2    0  249    0    0    1    2   13   20
##      [4,]    0    1    0   72    5    0    0    0    0
##      [5,]    0    0    0    0  114    0    8    4    0
##      [6,]    0    0    0    0    0   20    0    0    0
##      [7,]   18    0    2    0   11    0   77    8    3
##      [8,]    0    0    8    0    4    0    6  131   22
##      [9,]    0    0    1    0    0    0    0    0   34
```

```r
# Convert the predicted values to a factor variable that matches the factor levels in the actual values
tree.pred_factor <- factor(tree.pred, levels = levels(test$sp))

# Check the length of the predicted values and the actual values
#cat("Length of predicted values:", length(tree.pred_factor), "\n")
#cat("Length of actual values:", length(test$sp), "\n")

# Calculate the misclassification rate
misclassification <- mean(tree.pred_factor != test$sp)

# Print the misclassification rate
cat("Misclassification rate:", misclassification, "\n")
```

```
## Misclassification rate: 0.1645857
```

```r
set.seed(10)
cv.class<-cv.tree(tree.class, FUN = prune.misclass, K=10)
plot(cv.class$size, cv.class$dev,type="b")
```


```
prune.class=prune.tree(tree.class,best=17)
plot(prune.class)
text(prune.class,pretty=0,srt=45,cex=0.5)
```




```
# Make predictions on the test dataset using the pruned decision tree
prune.pred <- predict(prune.class, test, type = "class")
t=table(prune.pred,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL
t
```

```
##
## prune.pred [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##       [1,]    6    0    0    0    0    0    4    0    0
##       [2,]    0   33    0    1    0    0    0    0    1
##       [3,]    2    0  249    0    0    1    2   13   20
##       [4,]    0    1    0   72    5    0    0    0    0
##       [5,]    0    0    0    0  114    0    8    4    0
##       [6,]    0    0    0    0    0   20    0    0    0
##       [7,]   16    0    2    0    8    0   61    5    3
##       [8,]    2    0    8    0    7    0   22  111   14
##       [9,]    0    0    1    0    0    0    0   23   42
```

```
# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(prune.pred, levels = levels(test$sp))

# Calculate the misclassification rate
misclassification <- mean(prune.pred_factor != test$sp)
```

```
# Print the misclassification rate
cat("Misclassification rate:", misclassification, "\n")
```

```
## Misclassification rate: 0.1963678
```

```
misclass_tree<-function(idx){
train=df[-idx,]
test=df[idx,]
tree.class<-tree(factor(sp)~., train)
prune.class=prune.tree(tree.class,best=17)
prune.pred <- predict(prune.class, test, type = "class")

# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(prune.pred, levels = levels(test$sp))

# Calculate the misclassification rate
misclassification <- mean(prune.pred_factor != test$sp)

return(misclassification)
}
mis_tree<-lapply(folds,misclass_tree)
print(1-mean(as.numeric(mis_tree)))
```

```
## [1] 0.8073291
```

The accuracy for sp-male is 0.8073291.

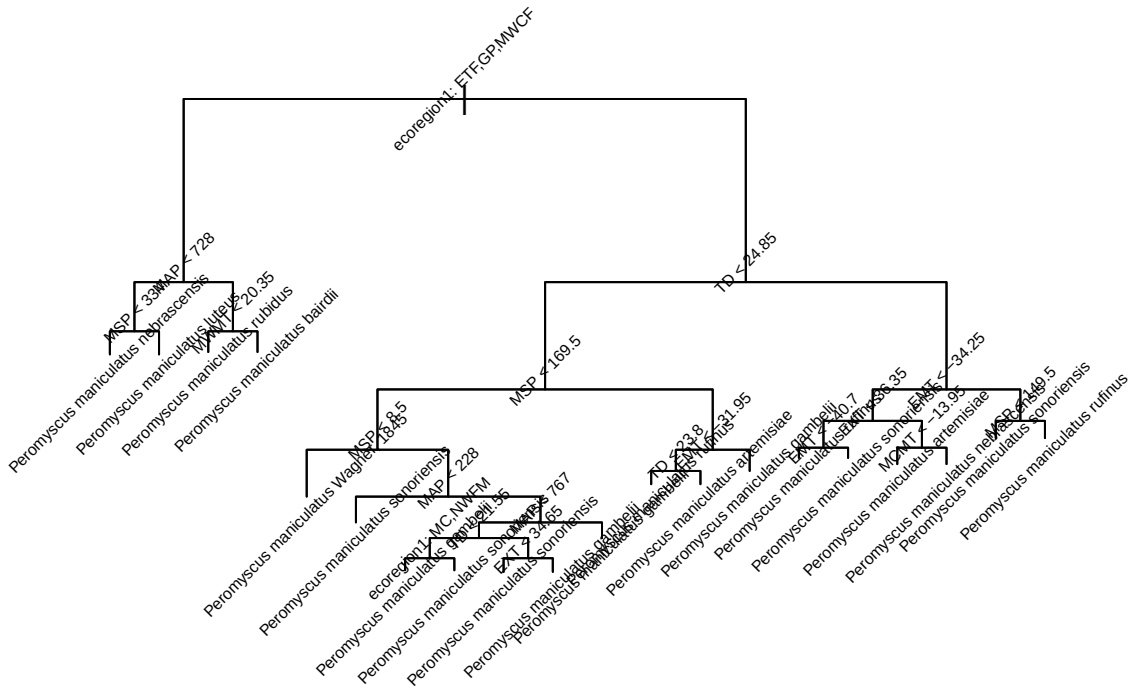
```
male <- subset(df, sex == "female")
set.seed(10)
idx=sample(1:nrow(male),3/4*nrow(male))
train=male[idx,]
test=male[-idx,]
```

```
tree.class<-tree(factor(sp)~., train)
summary(tree.class)
```

sp-female

```
##
## Classification tree:
## tree(formula = factor(sp) ~ ., data = train)
## Variables actually used in tree construction:
## [1] "ecoregion1" "MAP"        "MSP"        "MWMT"       "TD"
## [6] "EXT"        "EMT"        "MCMT"
## Number of terminal nodes:  20
## Residual mean deviance:  0.9433 = 1838 / 1949
## Misclassification error rate: 0.1498 = 295 / 1969
```

```
plot(tree.class)
text(tree.class,pretty=0,srt=45,cex=0.5)
```




```
# Make predictions on the test dataset using the pruned decision tree
tree.pred <- predict(tree.class, test, type = "class")
t=table(tree.pred,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL
t
```

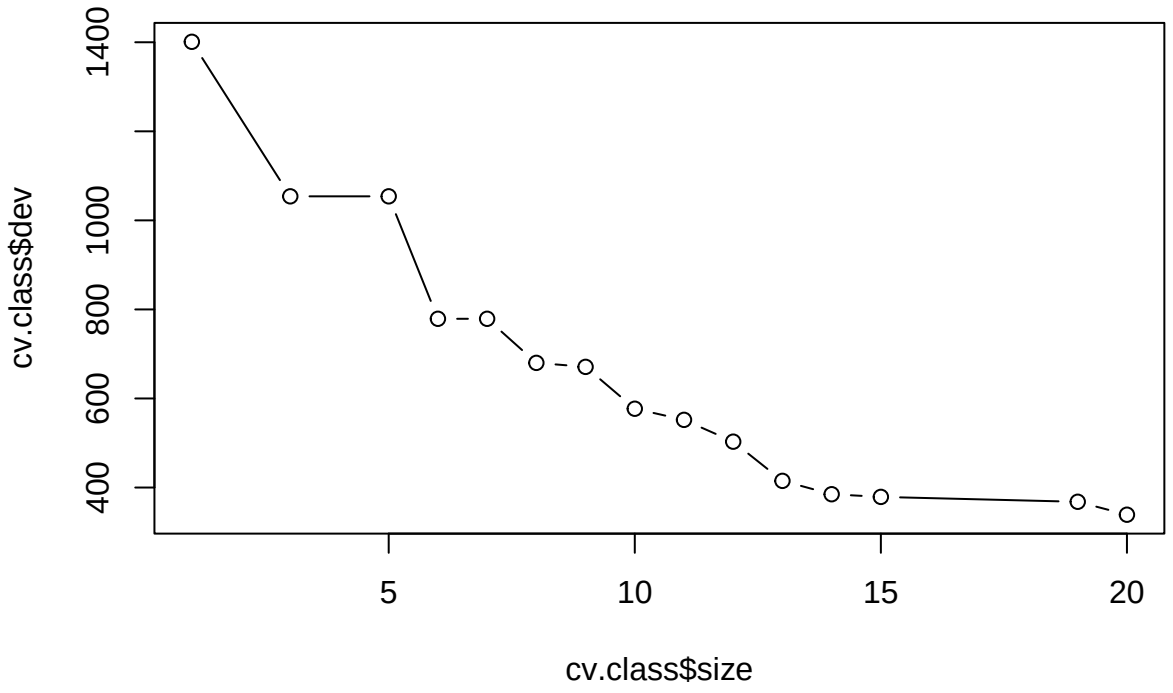
```
##
## tree.pred [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##       [1,]   15    0    0    0    0    0    0    0    1
##       [2,]    0   27    0    1    0    0    0    0    0
##       [3,]    1    0  152    0    3    1    6   15   18
##       [4,]    0    1    0   43    0    0    0    0    0
##       [5,]    0    1    0    3   88    0   12    2    0
##       [6,]    0    0    0    0    0   16    0    0    0
##       [7,]    8    0    0    0    7    0   57    6    0
##       [8,]    1    0    8    0    1    0   11  109   11
##       [9,]    0    0    0    0    0    0    0    0   32
```

```
# Re-level the predicted values to match the factor levels of the actual values
tree.pred_factor <- factor(tree.pred, levels = levels(test$sp))
```

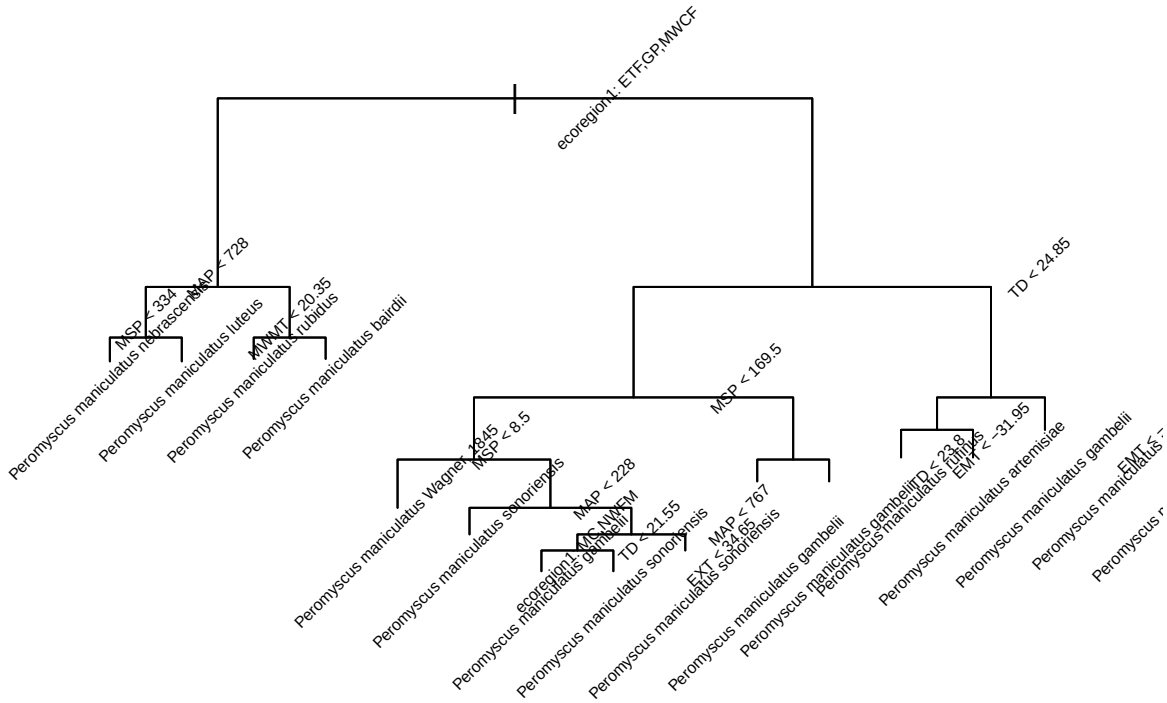
```r
# Calculate the misclassification rate
misclassification <- mean(tree.pred_factor != test$sp)

# Print the misclassification rate
cat("Misclassification rate:", misclassification, "\n")
```

```
## Misclassification rate: 0.1796043
```

```r
set.seed(10)
cv.class<-cv.tree(tree.class, FUN = prune.misclass, K=10)
plot(cv.class$size, cv.class$dev,type="b")
```




```r
prune.class=prune.tree(tree.class,best=14)
plot(prune.class)
text(tree.class,pretty=0,srt=45,cex=0.5)
```




```
# Make predictions on the test dataset using the pruned decision tree
prune.pred <- predict(prune.class, test, type = "class")
t=table(prune.pred,test$sp)
rownames(t) <- NULL
colnames(t) <- NULL
t
```

```
##
## prune.pred [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##       [1,]    0    0    0    0    0    0    0    0    0
##       [2,]    0   27    0    1    0    0    0    0    0
##       [3,]    1    0  154    0    3    1    7   41   23
##       [4,]    0    1    0   43    0    0    0    0    0
##       [5,]    9    1    0    3   88    0   12    2    0
##       [6,]    0    0    0    0    0   16    0    0    0
##       [7,]    7    0    0    0    7    0   42    6    1
##       [8,]    8    0    6    0    1    0   25   83    6
##       [9,]    0    0    0    0    0    0    0    0   32
```

```
# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(prune.pred, levels = levels(test$sp))

# Calculate the misclassification rate
misclassification <- mean(prune.pred_factor != test$sp)
```

```r
# Print the misclassification rate
cat("Misclassification rate:", misclassification, "\n")
```

```
## Misclassification rate: 0.261796
```

```r
misclass_tree<-function(idx){
train=df[-idx,]
test=df[idx,]
tree.class<-tree(factor(sp)~., train)
prune.class=prune.tree(tree.class,best=14)
prune.pred <- predict(prune.class, test, type = "class")

# Re-level the predicted values to match the factor levels of the actual values
prune.pred_factor <- factor(prune.pred, levels = levels(test$sp))

# Calculate the misclassification rate
misclassification <- mean(prune.pred_factor != test$sp)

return(misclassification)
}
mis_tree<-lapply(folds,misclass_tree)
print(1-mean(as.numeric(mis_tree)))
```

```
## [1] 0.7752934
```

The accuracy for sp-female is 0.7752934 compared with 0.8073291 in sp-male, the model for sp-male can do a better prediction.

6.3. Multinomial logistic regression (Author: Maciej Pecak)

Let's read the data first.

```
##     X.1     X       long      lat decade pop_density_4km2 month year season
## 1 18357 28224  -68.29874 44.33347   1950       4.31576252    11 1949   fall
## 2 16217 24353 -118.30000 34.53000   1970       0.07396096    10 1972   fall
## 3 17431 26512 -118.99695 34.78537   2000       0.00000000    11 2002   fall
##   lifestage                                    sp    sex body_mass tail_length
## 1     YOUNG    Peromyscus maniculatus abietorum   male      18.5          79
## 2        AD Peromyscus maniculatus Wagner, 1845   male      21.0          72
## 3        AD  Peromyscus maniculatus sonoriensis female      17.0          66
##   total_length HB.Length  MAT MWMT MCMT   TD  MAP MSP  DD5 FFP   EMT  EXT
## 1          151        72  8.4 21.2 -2.8 24.0 1136 374 2243 153 -32.7 44.1
## 2          164        92 15.7 26.0  6.5 19.5  131  18 3990 275 -10.3 43.7
## 3          155        89  8.9 18.4  1.2 17.2  411  15 1929 136 -24.5 37.5
##                   ecoregion1 ecoregion1_num season_num sex_num sex_transformed
## 1 EASTERN TEMPERATE FORESTS            1.0          3       1               1
## 2  MEDITERRANEAN CALIFORNIA            6.5          3       1               1
## 3  MEDITERRANEAN CALIFORNIA            6.5          3       0               0
##   ecoregion1_transformed season_transformed
```

```
## 1                     1                 2
## 2                     6                 2
## 3                     6                 2
```

For the purpose of better classification and consistency with other algorithms, only mice subspecies that have more than a hundred observations will be included.

```
species.considered <- c(
  "Peromyscus maniculatus Wagner, 1845",
  "Peromyscus maniculatus sonoriensis",
  "Peromyscus maniculatus bairdii",
  "Peromyscus maniculatus gambelii",
  "Peromyscus maniculatus artemisiae",
  "Peromyscus maniculatus rufinus",
  "Peromyscus maniculatus rubidus",
  "Peromyscus maniculatus nebrascensis",
  "Peromyscus maniculatus luteus"
)

model.df <- mice.df %>%
  select(-c(X, X.1, long, lat, decade, month, year, ecoregion1_num,
            season_num, sex_num, sex_transformed,
            ecoregion1_transformed, season_transformed)) %>%
  filter(sp %in% species.considered)
```

In order to ensure the correctness of the algorithm, the variables that are highly multicolinear need to be eliminated. Is was conducted manually by eliminating the correlated variables one by one, based on the highest value of the Variance Inflation Factor. The following subset contains values that are not colinear.

```
vif(lm(body_mass ~
         pop_density_4km2 + tail_length + HB.Length +
         TD + MAP + MSP + FFP + EXT, data = model.df))
```

```
## pop_density_4km2      tail_length        HB.Length               TD
##         1.060649         1.368813         1.166767         2.424823
##              MAP              MSP              FFP              EXT
##         1.884722         2.405276         2.651601         3.369802
```

The next step is to divide the data into train and test sets in 3:1 proportion.

```
set.seed(10)
model.df <- model.df %>%
  mutate(id = 1:nrow(model.df))

model.train <- model.df %>% sample_frac(.75)
model.test <- anti_join(model.df, model.train, by = "id")

model.train <- dplyr::select(model.train, -id)
model.test <- dplyr::select(model.test, -id)
```

Finally, the model that uses all non-correlated numerical variables as well as available categorical variables. Since different species cannot be ordered, the only reasonable approach is to use the baseline probability model.

```
species.baseline.classifier <- vglm(
  sp ~ pop_density_4km2 + tail_length + HB.Length + TD + MAP + MSP + FFP + EXT,
  family = multinomial,
  data = model.train
)
```

A basic diagnostic to check whether the trained model is reasonable (χ^2-goodness of fit test)

```
1-pchisq(
  deviance(species.baseline.classifier),
  df.residual(species.baseline.classifier)
)
```

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

As shown above, the p-value for the goodness-of-fit test fails to reject the null hypothesis.

Finally, let's compute the correct classification rate (accuracy).

```
predicted.vals <- predict(species.baseline.classifier, model.test, type="response")
fitted.result<-colnames(predicted.vals)[rowMaxs(predicted.vals)]

misClasificError <- mean(fitted.result != model.test$sp)
print(paste('Accuracy:', 1 - misClasificError))
```

```
## [1] "Accuracy: 0.754226267880364"
```

Multinomial Logistic Regression - Cross Validation

For the purpose of comparison between the other classification methods, the 10-fold cross validation needs to be conducted. As a performance measure, the cross validation error is defined as the misclassification rate.

```
library("caret")
set.seed(10)
no.folds = 10
folds <- createFolds(model.df$sp, k = no.folds)
```

For each fold, the model is trained on the remaining ones and the misclassification rate is computed on the currently selected fold.

```
cv.errors <- numeric(no.folds)
for(i in 1:no.folds) {
  train <- model.df[-unlist(folds[i]), ]
  test <- model.df[unlist(folds[i]), ]

  species.baseline.cv.classifier <- vglm(
    sp ~ pop_density_4km2 + tail_length + HB.Length + TD + MAP + MSP + FFP + EXT,
    family = multinomial,
```

```
    data = train
  )

  predicted.vals <- predict(species.baseline.cv.classifier, test, type="response")
  fitted.result <- colnames(predicted.vals)[rowMaxs(predicted.vals)]

  cv.errors[i] <- mean(fitted.result != test$sp)
}

cv.errors
```

```
##  [1] 0.2822186 0.2985318 0.2560778 0.3034258 0.2662338 0.2687296 0.2605178
##  [8] 0.2861789 0.2585366 0.2678571
```

Final accuracy based on the cross validation.

```
1 - mean(cv.errors)
```

```
## [1] 0.7251692
```

6.4. Classification methods - comparison (Authors: Maciej Pecak, Hao Su)

The following table summarizes explored classification methods for the deer mouse subspecies.

Method	Met assumptions	CV Error
LDA	😉	0.7119
QDA	😉	0.7099
Decision Tree	👍	0.7992
Multinomial Logistic Regression	👍	0.7252

To sum up, the classification methods performed at comparable level. The LDA and QDA shouldn't be used for further prediction as the normality and homoscedasticity (for LDA) assumptions were not met. The best performance could be obtained with the decision tree.

Summary

For this project we explored and practised techniques that were presented in the class:

- Practiced kNN algorithm by using it to fill missing data (lifestage - categorical kNN, body mass - bootstrapping nearest neighbors).
- Conducted analysis to deremine whether the 30 years time periods and ecoregions should be treated as strata or clusters when estimating body mass, tail length and head-body length. Compared the values of SSB and SS and performed Kruskal-Wallis test since the homoscedasticity assumption was not met for ANOVA. In both cases.
- Estimated population body mass and tail length using simple random sampling and stratified sampling. Cluster sampling was rejected in the previous analysis. In both cases, stratified sampling yielded smaller standard error compared to SRS.
- Compared linear regression with regression tree for predicting body mass and total length.
- Explored the impact of climate variables on body mass and total length using linear regression and regression trees.
- Compared four different methods of classification: LDA, QDA, Classification tree and Multinomial Logistic Regression to classify deer mice subspecies.

References

- Guralnick, Robert, i in. „Body Size Trends in Response to Climate and Urbanization in the Widespread North American Deer Mouse, Peromyscus Maniculatus". Scientific Reports, t. 10, nr 1, June 2020, s. 8882. www.nature.com, https://doi.org/10.1038/s41598-020-65755-x.
- Wang T, Hamann A, Spittlehouse D, Carroll C (2016) Locally Downscaled and Spatially Customizable Climate Data for Historical and Future Periods for North America. PLoS ONE 11(6): e0156720. doi:10.1371/journal.pone.0156720
- Mahony CR, Wang T; Hamann A and Cannon AJ, 2022. A CMIP6 ensemble for downscaled monthly climate normals over North America. EarthArXiv. https://doi.org/10.31223/X5CK6Z